



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY  
A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

**ÚSTAV TELEKOMUNIKACÍ**

DEPARTMENT OF TELECOMMUNICATIONS

**INTERAKTIVNÍ WEBOVÉ APLIKACE PRO PODPORU  
VÝUKY ZPRACOVÁNÍ SIGNÁLŮ**

INTERACTIVE WEB APPLICATIONS FOR SUPPORT OF TEACHING SIGNAL PROCESSING

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Petr Mašek**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. Pavel Závíška**

**BRNO 2019**

# Bakalářská práce

bakalářský studijní obor **Teleinformatika**

Ústav telekomunikací

**Student:** Petr Mašek

**ID:** 197645

**Ročník:** 3

**Akademický rok:** 2018/19

## NÁZEV TÉMATU:

**Interaktivní webové aplikace pro podporu výuky zpracování signálů**

## POKYNY PRO VYPRACOVÁNÍ:

V jazyce JavaScript vytvořte čtyři webové aplikace, které poslouží pro interaktivní podporu výuky zejména v kurzech týkajících se zpracování signálů.

Aplikace budou tématicky zaměřeny na:

- 1) převzorkování diskrétního signálu v poměru racionálního čísla,
- 2) střední a efektivní hodnotu signálu,
- 3) diskrétní lineární a kruhovou konvoluci v 1D,
- 4) diskrétní křížovou korelaci.

Zaměřte se především na názornou podobu a funkčnost pro potřebu výuky.

## DOPORUČENÁ LITERATURA:

[1] Smékal, Z.: Analýza signálu a soustav. Vysoké učení technické v Brně, 2012. ISBN 978-80-214-4453-9.

[2] Smith, S. W.: The scientist and engineer's guide to digital signal processing. San Diego, Calif.: California Technical Pub., 1997. ISBN 09-660-1763-3.

**Termín zadání:** 1.2.2019

**Termín odevzdání:** 27.5.2019

**Vedoucí práce:** Ing. Pavel Záviška

**Konzultant:**

**prof. Ing. Jiří Mišurec, CSc.**  
*předseda oborové rady*

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## ABSTRAKT

Bakalářská práce se zabývá problematikou zpracování jednorozměrných signálů. V rámci práce bylo vytvořeno pět interaktivních webových aplikací, naprogramovaných v JavaScriptu, které mají sloužit pro podporu výuky. Jedná se o applety, které se týkají následujících témat: převzorkování diskrétního signálu v poměru racionálního čísla, střední a efektivní hodnota signálu, diskrétní lineární a kruhová konvoluce v 1D a diskrétní křížová korelace. V práci je ke každému tématu věnován teoretický úvod, popis implementace a uživatelského rozhraní vytvořených aplikací. Dále je v práci uveden komentář k vývoji a testování appletů.

## KLÍČOVÁ SLOVA

převzorkování, střední hodnota, efektivní hodnota, lineární konvoluce, kruhová konvoluce, křížová korelace, signál, applet

## ABSTRACT

The bachelor thesis deals with the topic of processing one-dimensional signals. Five interactive web applications have been created in JavaScript to support teaching. These are applets that relate to the following topics: rational number resampling of the discrete signal, mean and root mean square signal values, discrete linear and circular convolution in 1D, and discrete cross correlation. The thesis introduces each topic by a theoretical foreword, followed by description of implementation and user interface of the created applications and further there is a commentary on the development and testing of applets.

## KEYWORDS

resampling, mean value, root mean square, linear convolution, circular convolution, cross correlation, signal, applet

MAŠEK, Petr. *Interaktivní webové aplikace pro podporu výuky zpracování signálů*. Brno, 2019, 52 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Pavel Závíška

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Interaktivní webové aplikace pro podporu výuky zpracování signálů“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Pavlu Záviškovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno .....

.....

podpis autora

# Obsah

Úvod	9
<b>1 Teorie k apletu Převzorkování diskrétního signálu v poměru racionálního čísla</b>	<b>10</b>
1.1 Vzorkování . . . . .	10
1.2 Podvzorkování diskrétního signálu . . . . .	11
1.3 Nadvzorkování diskrétního signálu . . . . .	12
1.4 Převzorkování diskrétního signálu . . . . .	14
<b>2 Teorie k apletu Střední a efektivní hodnota signálu</b>	<b>17</b>
2.1 Střední hodnota signálu . . . . .	17
2.2 Efektivní hodnota signálu . . . . .	18
2.3 Poměrné činitele . . . . .	19
<b>3 Teorie k apletu Diskrétní lineární a kruhová konvoluce v 1D</b>	<b>21</b>
3.1 Lineární konvoluce . . . . .	21
3.2 Diskrétní lineární konvoluce . . . . .	21
3.3 Diskrétní kruhová konvoluce . . . . .	24
<b>4 Teorie k apletu Diskrétní křížová korelace</b>	<b>26</b>
4.1 Křížová korelace . . . . .	26
4.2 Diskrétní křížová korelace . . . . .	26
<b>5 Vývoj apletů</b>	<b>29</b>
5.1 Použité jazyky . . . . .	29
5.1.1 JavaScript . . . . .	29
5.1.2 HTML . . . . .	31
5.1.3 CSS . . . . .	31
5.2 Testování . . . . .	32
<b>6 Popis vytvořených apletů</b>	<b>34</b>
6.1 Převzorkování diskrétního signálu v poměru racionálního čísla . . . .	34
6.2 Střední a efektivní hodnota signálu . . . . .	37
6.3 Diskrétní lineární a kruhová konvoluce v 1D . . . . .	39
6.3.1 Diskrétní lineární konvoluce v 1D . . . . .	39
6.3.2 Diskrétní kruhová konvoluce v 1D . . . . .	42
6.4 Diskrétní křížová korelace . . . . .	45
<b>7 Závěr</b>	<b>48</b>

Literatura	49
Seznam symbolů, veličin a zkratk	51
A Obsah přiloženého CD	52

# Seznam obrázků

1.1	Druhy vzorkování . . . . .	11
1.2	Příklad podvzorkování s činitelem podvzorkování $D = 2$ . . . . .	13
1.3	Příklad nadvzorkování s činitelem nadvzorkování $L = 3$ . . . . .	14
1.4	Příklad převzorkování s činitelem nadvzorkování $L = 4$ a činitelem podvzorkování $D = 3$ . . . . .	16
2.1	Výpočet střední hodnoty signálu $i(t)$ . . . . .	18
2.2	Výpočet efektivní hodnoty signálu $i(t)$ . . . . .	19
3.1	Příklad lineární konvoluce $y(t) = g(t) * x(t)$ . . . . .	22
3.2	Příklad diskrétní lineární konvoluce $y[n] = h[n] * x[n]$ . . . . .	23
3.3	Příklad diskrétní kruhové konvoluce $y[n] = h[n] \otimes x[n]$ . . . . .	25
4.1	Příklad křížové korelace $y(t) = g(t) \star x(t)$ . . . . .	27
4.2	Příklad diskrétní křížové korelace $y[n] = h[n] \star x[n]$ . . . . .	28
5.1	Vývojové prostředí WebStorm . . . . .	29
5.2	Ukázka chování apletu, pokud není povolen JavaScript . . . . .	32
6.1	Aplet Převzorkování diskrétního signálu v poměru racionálního čísla . . . . .	36
6.2	Aplet Střední a efektivní hodnota signálu . . . . .	38
6.3	Aplet Diskrétní lineární konvoluce v 1D . . . . .	41
6.4	Aplet Diskrétní kruhová konvoluce v 1D . . . . .	44
6.5	Aplet Diskrétní křížová korelace . . . . .	47



# Úvod

Cílem bakalářské práce je vytvoření webových aplikací, které mají sloužit pro podporu výuky zejména v kurzech týkajících se zpracování signálů. Konkrétně může jít o předmět Analýza signálů a soustav (BASS), který je součástí 2. ročníku oborů Teleinformatika, Audio inženýrství, Mikroelektronika a technologie, respektive předmět Analýza signálů a soustav (BPC-ASI), který je součástí 2. ročníku oborů Telekomunikační a informační systémy, Audio inženýrství, Mikroelektronika a technologie v bakalářském studiu. Nebo může jít o předmět Číslicové zpracování signálů (MCSI), který je součástí 1. ročníku oboru Telekomunikační a informační technika v navazujícím magisterském studiu na Fakultě elektrotechniky a komunikačních technologií Vysokého učení technického v Brně.

V rámci bakalářské práce tedy bylo vytvořeno pět interaktivních apletů v jazyce JavaScript, které přibližují problematiku týkající se následujících témat: převzorkování diskrétního signálu v poměru racionálního čísla, střední a efektivní hodnota signálu, diskrétní lineární a kruhová konvoluce v 1D a diskrétní křížová korelace.

První kapitola se týká teoretického úvodu k apletu Převzorkování diskrétního signálu v poměru racionálního čísla. Popsán je princip vzorkování, podvzorkování, nadvzorkování a převzorkování signálů.

V druhé kapitole se pojednává o teoretickém základu k apletu Střední a efektivní hodnota signálu. Kromě matematického popisu střední a efektivní hodnoty je v kapitole zmíněno využití tzv. poměrných činitelů.

Ve třetí kapitole je popsáno, jakým způsobem lze vypočíst lineární a kruhovou konvoluci signálů. Tato teoretická část slouží k vysvětlení principů apletů Diskrétní lineární a kruhová konvoluce v 1D.

Čtvrtá kapitola pojednává o matematickém popisu křížové korelace. Ten je zmíněn pro lepší pochopení fungování apletu Diskrétní křížová korelace.

Pátá kapitola se věnuje vývoji jednotlivých apletů. Jsou zde vyjmenovány a popsány programovací jazyky, které byly využity pro vývoj webových aplikací. Dále je zde zmíněno, jak probíhalo testování apletů.

V poslední, šesté kapitole je popsána funkcionalita a popis ovládání jednotlivých webových aplikací.

Přáním autora je, aby vytvořené aplety posloužily pro podporu výuky signálů. Studenti by si tak mohli snadno a rychle ověřit, zda jsou jejich výpočty správné. Pomocí interaktivních grafů mohou studenti také lépe porozumět problematice dané látky. Pozorovat mohou, jaké změny se dějí na výstupu, když se pozvolna mění vstupní hodnoty.

# 1 Teorie k apletu Převzorkování diskrétního signálu v poměru racionálního čísla

Následující kapitola popisuje proces vzorkování, podvzorkování, nadvzorkování a převzorkování signálů.

## 1.1 Vzorkování

Vzorkování signálu je proces, při kterém se časově spojitý signál převádí na diskrétní. Z časově spojitého signálu se zachová pouze několik původních hodnot – vzorků. Vzorky jsou odebírány v pravidelných intervalech – vzorkovací perioda  $T_{vz}$ . Převrácenou hodnotou vzorkovací periody je vzorkovací frekvence  $f_{vz}$ , pro kterou platí [18]:

$$f_{vz} = \frac{1}{T_{vz}}, \quad (1.1)$$

která udává počet vzorků za jednotku času.

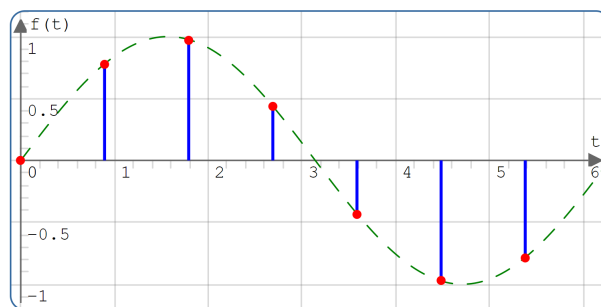
Vzorkování (sampling) lze rozdělit do tří skupin: ideální vzorkování, vzorkování prvního druhu a vzorkování druhého druhu. Ideální vzorkování je matematický model, kdy jsou vzorky odebírány v časových bodech s nulovou délkou (viz obrázek 1.1a). Při vzorkování prvního druhu se z původního analogového signálu vybírají pouze krátké časové úseky (viz obrázek 1.1b). Vzorkování druhého druhu spočívá v tom, že je původní signál nahrazen pravoúhlými impulzy stejného tvaru s výškou odpovídající velikosti signálu v okamžiku vzorkování (viz obrázek 1.1c) [18, 5].

Pro správné vzorkování musí být dodržena vzorkovací poučka (označováno také jako Shannon-Kotelnikovův teorém, Nyquistovo kritérium a podobně) [18]:

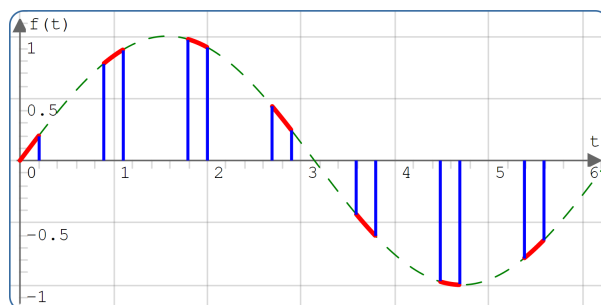
$$f_{vz} > 2f_{\max}, \quad (1.2)$$

kde  $f_{vz}$  je vzorkovací frekvence a  $f_{\max}$  je maximální frekvence složky spektra zkoumaného signálu. Pokud tato podmínka nebude dodržena, dojde při zpětné rekonstrukci k tzv. aliasingu – deformaci původního signálu z důvodu překrývajících se sousedních spekter signálu [18].

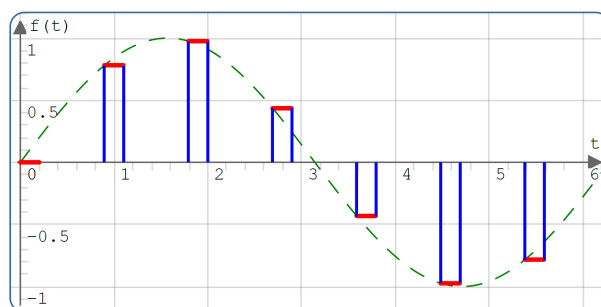
U digitalizace signálů se kromě vzorkování užívá také kvantování a kódování. Kvantování spočívá v diskretizaci naměřených hodnot (podobně jako diskretizace časového průběhu u vzorkování). Při kvantování však vzniká kvantovací šum, který nenávratně zkreslí původní hodnoty. Kódování přiřazuje jednotlivým hladinám diskretizovaných hodnot jedinečné číselné vyjádření s využitím nejčastěji dvojkové soustavy. Proces, který zahrnuje všechny tři operace – vzorkování, kvantování, kódování (v uvedeném pořadí) – se nazývá analogově číslicový (analogově digitální A/D) převod [18, 7].



(a) Ideální vzorkování.



(b) Vzorkování prvního druhu.



(c) Vzorkování druhého druhu.

Obr. 1.1: Druhy vzorkování.

## 1.2 Podvzorkování diskrétního signálu

Podvzorkování (downsampling, decimace) diskrétního signálu v poměru celého čísla je postup, kdy se původní vzorkovací frekvence signálu zmenší tím, že se vypustí určité vzorky. Toho se dosáhne tím, že se zvýší vzorkovací perioda [19, 20].

Míru podvzorkování určuje činitel (faktor) podvzorkování  $D$ . Udává, že výsledný podvzorkovaný signál bude mít  $D$ -krát méně vzorků než původní signál. Doba periody bude naopak  $D$ -krát větší. Podvzorkování si lze také představit, že po jednom

zachovaném vzorku z původního signálu se  $D - 1$  vzorků odebere [19, 20, 11].

Výsledný kmitočet podvzorkovaného signálu je [19, 20]:

$$f_{vz_n} = \frac{1}{D} f_{vz}, \quad (1.3)$$

kde  $f_{vz_n}$  je nová vzorkovací frekvence,  $D$  je činitel podvzorkování a  $f_{vz}$  je původní vzorkovací frekvence.

Aby však nedošlo k aliasingu, musí být splněna vzorkovací poučka, která lze zapsat také ve tvaru [19, 20]:

$$f_{\max} < \frac{1}{2} f_{vz_n}, \quad (1.4)$$

kde  $f_{\max}$  je maximální frekvence složky spektra původního signálu a  $f_{vz_n}$  je nová vzorkovací frekvence po podvzorkování.

Aby byla jistota, že je vzorkovací teorém (1.4) splněn, bývá před samotnou operací podvzorkování zpravidla přidáván (antialiasingový) filtr typu dolní propust s mezním kmitočtem  $f_{DP} < \frac{1}{2} f_{vz_n}$ , čímž se omezí kmitočtové spektrum původního signálu [19, 20, 7].

Výstupní podvzorkovaný signál  $y[n]$  je možné vyjádřit jako [19, 20]:

$$y[n] = h[n] * x[nD] = \sum_{m=-\infty}^{\infty} h[m] x[nD - m], \quad (1.5)$$

kde  $h[n]$  je impulzní odezva antialiasingového filtru, operátor  $*$  značí operaci lineární konvoluce,  $x[nD]$  je podvzorkovaný původní diskretní signál, kde  $n$  značí index vzorku,  $D$  je činitel podvzorkování a  $m$  je iterátor sumy.

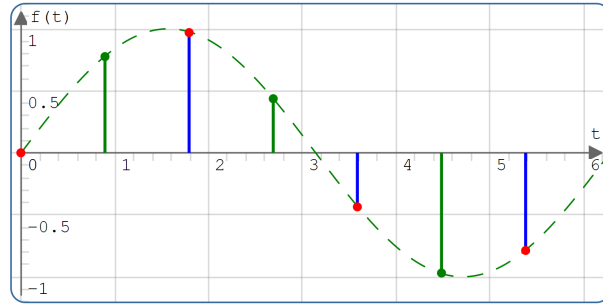
Příklad podvzorkování je na obrázku 1.2.

## 1.3 Nadvzorkování diskretního signálu

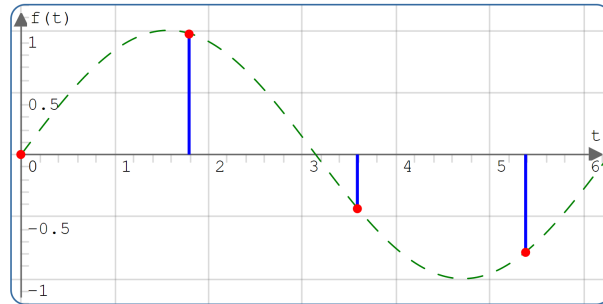
Nadvzorkování (upsampling, expanze, interpolace) diskretního signálu v poměru celého čísla je postup, kdy se původní vzorkovací frekvence zvýší tím, že se vzorkuje s nižší vzorkovací periodou. Mezi původními vzorky, které zůstávají zachované, se tak vloží nové vzorky [19, 20].

Míru nadvzorkování určuje činitel (faktor) nadvzorkování  $L$ . Udává, že výsledný nadvzorkovaný signál bude mít  $L$ -krát více vzorků než původní signál. Doba periody bude naopak  $L$ -krát menší. Nadvzorkování si lze také představit, že se mezi vzorky původního signálu vloží  $L - 1$  nových vzorků [19, 20, 11].

V prvním kroku jsou však tyto nově vložené vzorky nulové, protože v daných místech není definovaná hodnota. Proto je součástí procesu nadvzorkování filtrace dolní propustí, která aproximuje nově vložené nulové vzorky na požadované hodnoty [19, 20].



(a) Původní signál.



(b) Podvzorkovaný signál.

Obr. 1.2: Příklad podvzorkování s činitelem podvzorkování  $D = 2$ .

Výsledný nadvzorkovaný signál má frekvenci [19, 20]:

$$f_{vzn} = L f_{vz}, \quad (1.6)$$

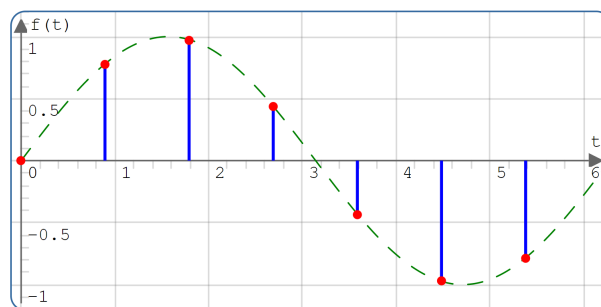
kde  $f_{vzn}$  je nová vzorkovací frekvence,  $L$  je činitel nadvzorkování a  $f_{vz}$  je původní vzorkovací frekvence.

Výstupní nadvzorkovaný signál  $y[n]$  je možné vyjádřit jako [19, 20]:

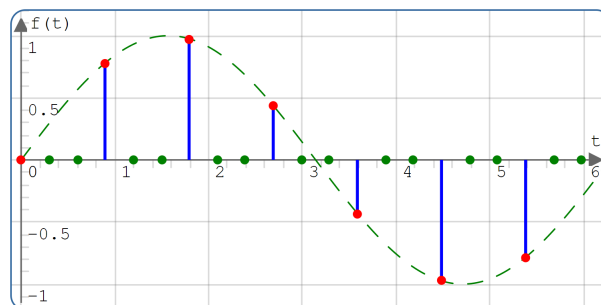
$$y[n] = h[n] * x \left[ \frac{1}{L}n \right] = \sum_{m=-\infty}^{\infty} h[m] x \left[ \frac{1}{L}n - m \right], \quad (1.7)$$

kde  $h[n]$  je impulzní odezva filtru typu dolní propust, operátor  $*$  značí operaci lineární konvoluce,  $x \left[ \frac{1}{L}n \right]$  je nadvzorkovaný původní diskretní signál, kde  $n$  značí index vzorku,  $L$  je činitel nadvzorkování a  $m$  je iterátor sumy.

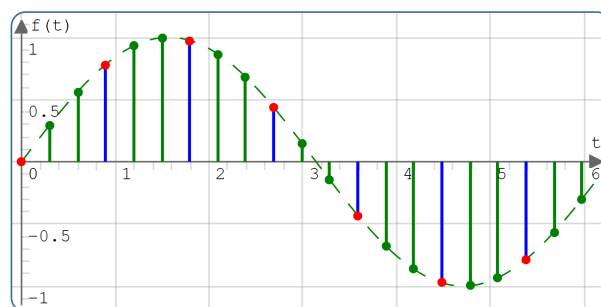
Příklad nadvzorkování je na obrázku 1.3.



(a) Původní signál.



(b) Přidání  $L - 1$  nulových vzorků.



(c) Nadvzorkovaný signál – po filtraci.

Obr. 1.3: Příklad nadvzorkování s činitelem nadvzorkování  $L = 3$ .

## 1.4 Převzorkování diskrétního signálu

Převzorkování diskrétního signálu v poměru racionálního čísla se v praxi hojně využívá. Například při konverzi hudebních záznamů pořizovaných v hudebních studiích, kde se používá vzorkovací frekvence 48 kHz, a jejich následné uložení na CD s kmitočtem 44,1 kHz. Převzorkování je v tomto případě výrazně lepší varianta, než převedení diskrétního signálu na analogový a následné navzorkování požadovanou vzorkovací frekvencí. Při tomto postupu by totiž došlo k řadě nepřesnostem a chyb,

které by zkreslily signál natolik, že by nebyl použitelný [19, 20].

U celkového nadvzorkování či podvzorkování v poměru racionálního čísla se využívá toho, že racionální číslo lze nahradit podílem dvou celých čísel. Proces převzorkování diskrétního signálu v poměru racionálního čísla se tak realizuje postupným nadvzorkováním (viz kapitola 1.3) a následným podvzorkováním (viz kapitola 1.2) diskrétního signálu v poměru celých čísel [19, 20, 23].

Činitel pro nadvzorkování  $L$  a činitel pro podvzorkování  $D$  vychází z původní a požadované vzorkovací frekvence. Lze je vypočíst následovně [19, 20]:

$$L = \frac{f_{vz_n}}{\text{GCD}(f_{vz}, f_{vz_n})} \quad (1.8)$$

a

$$D = \frac{f_{vz}}{\text{GCD}(f_{vz}, f_{vz_n})}, \quad (1.9)$$

kde  $\text{GCD}(f_{vz}, f_{vz_n})$  značí největší společný dělitel původní vzorkovací frekvence  $f_{vz}$  a požadované vzorkovací frekvence  $f_{vz_n}$ . Výpočetní náročnost se po vydělení kmitočtů největším společným dělitelem se několikanásobně zjednoduší.

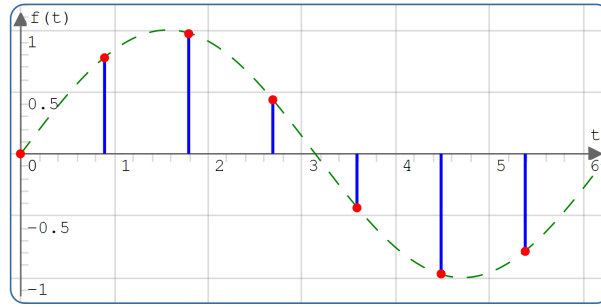
Výsledný vzorkovací kmitočet převzorkovaného signálu je [19, 20]:

$$f_{vz_n} = \frac{L}{D} f_{vz}, \quad (1.10)$$

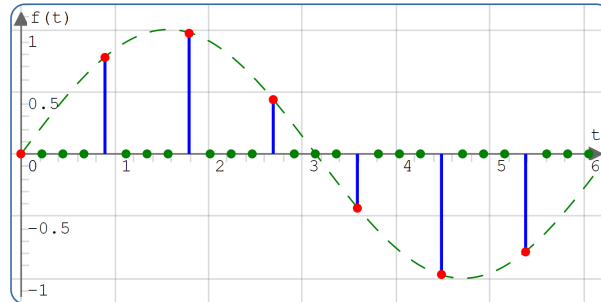
kde  $f_{vz_n}$  je nová vzorkovací frekvence,  $L$  je činitel nadvzorkování,  $D$  je činitel podvzorkování a  $f_{vz}$  je původní vzorkovací frekvence.

Pokud platí, že  $L > D$ , původní signál bude prakticky nadvzorkovaný. Naopak pokud  $D > L$ , výsledný signál bude podvzorkovaný. Pokud  $L = D$ , původní i výsledná vzorkovací frekvence bude totožná, převzorkování v tomto případě tedy postrádá smysl.

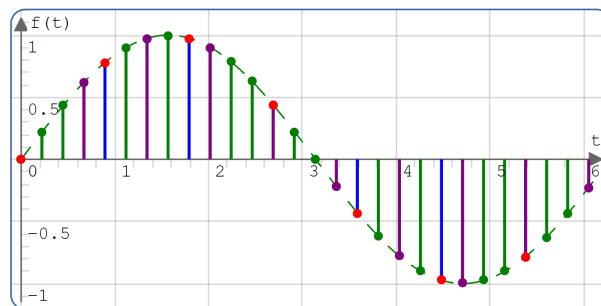
Příklad převzorkování je na obrázku 1.4.



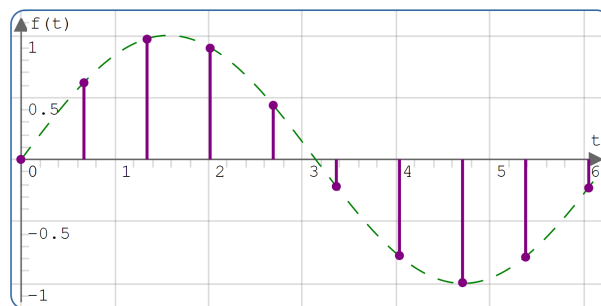
(a) Původní signál.



(b) Přidání  $L - 1$  nulových vzorků.



(c) Nadvzorkovaný signál – po filtraci.



(d) Převzorkovaný signál – po podvzorkování.

Obr. 1.4: Příklad převzorkování s činitelem nadvzorkování  $L = 4$  a činitelem podvzorkování  $D = 3$ .



## 2 Teorie k apletu Střední a efektivní hodnota signálu

Střední i efektivní hodnoty signálů jsou v elektrotechnice důležitými hodnotami. Jsou používány v řadě měřicích přístrojů, jako jsou například ampérmetry, voltmetry nebo multimetry. Popisují časově proměnnou veličinu pomocí konstantní funkce. Často se využívají pro popis periodických signálů.

Kromě elektrických signálů lze tyto hodnoty použít pro popis signálu elektromagnetického, optického nebo akustického.

### 2.1 Střední hodnota signálu

Střední hodnota  $I_s$  je stejnosměrná složka signálu, která udává průměrnou hodnotu funkce v době jedné periody [3].

Lze ji vypočítat vztahem [1, 3, 18]:

$$I_s = \frac{1}{T_1} \int_0^{T_1} i(t) dt, \quad (2.1)$$

kde  $T_1$  je základní perioda signálu,  $i(t)$  je časový průběh signálu,  $t$  je čas. U elektrického signálu představuje hodnotu stejnosměrného proudu, který za dobu jedné periody přenesel stejný celkový náboj [3].

Rovnici (2.1) lze zapsat i ve tvaru:

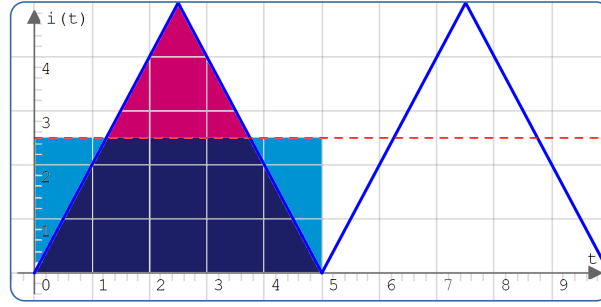
$$I_s T_1 = \int_0^{T_1} i(t) dt, \quad (2.2)$$

kde na levé straně představuje obsah obdélníku se stranami  $I_s$  a  $T_1$ . Na pravé straně rovnice se nachází integrál, který znázorňuje plochu mezi křivkou funkce  $i(t)$  a horizontální osou. Příklad výpočtu střední hodnoty s pomocí geometrické interpretace je na obrázku 2.1.

Pro střídavé signály, které mají během jedné periody stejnou plochu pod křivkou i nad křivkou, je výsledná střední hodnota nulová [1].

Pokud časový průběh funkce  $i(t)$  nelze popsat jednou matematickou funkcí, lze pro výpočet integrálu původní funkci rozdělit na dílčí podfunkce ( $i_1(t)$ ,  $i_2(t)$ , ...), spočítat jednotlivé integrály (s příslušnými mezemi) a ty potom sečíst. Například:

$$\int_0^{T_1} i(t) dt = \int_0^{T_0} i_1(t) dt + \int_{T_0}^{T_1} i_2(t) dt. \quad (2.3)$$



Obr. 2.1: Výpočet střední hodnoty signálu  $i(t)$ . Červeně znázorněna výsledná střední hodnota.

Dále je definována střední absolutní hodnota  $I_{sa}$  (podle [1, 3] aritmetická střední hodnota), která se od střední hodnoty liší v tom, že záporné hodnoty původní signálu se převedou na kladné [1].

$$I_{sa} = \frac{1}{T_1} \int_0^{T_1} |i(t)| dt. \quad (2.4)$$

## 2.2 Efektivní hodnota signálu

Efektivní hodnota  $I_{ef}$  je jedna z nejvýznamnějších integrálních hodnot [3]. Z fyzikálního hlediska se jedná o konstantní funkci, která má stejné energetické účinky, jako původní signál. Lze ji vypočítat vztahem [1, 3, 18]:

$$I_{ef} = \sqrt{\frac{1}{T_1} \int_0^{T_1} i^2(t) dt}, \quad (2.5)$$

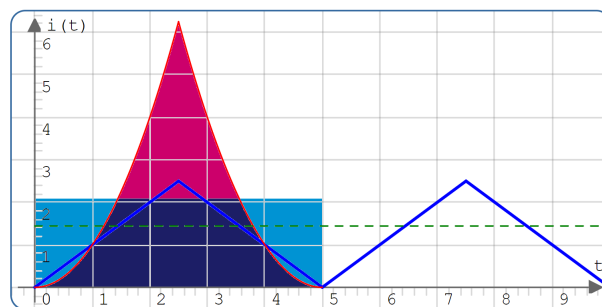
kde  $T_1$  je perioda signálu,  $i(t)$  je časový průběh signálu,  $t$  je čas.

Rovnici (2.5) lze po umocnění zapsat ve tvaru:

$$I_{ef}^2 T_1 = \int_0^{T_1} i^2(t) dt, \quad (2.6)$$

kde na levá strana představuje obsah obdélníku se stranami  $I_{ef}^2$  a  $T_1$ . Na pravé straně rovnice se nachází integrál, který znázorňuje plochu mezi křivkou funkce  $i^2(t)$  a horizontální osou. Příklad výpočtu efektivní hodnoty s pomocí geometrické interpretace je na obrázku 2.2.

Výpočet efektivní hodnoty je podobný výpočtu střední hodnoty s rozdílem, že je nejprve umocněn na druhou průběh funkce  $i(t)$  a nakonec je hodnota odmocněna [18].



Obr. 2.2: Výpočet efektivní hodnoty signálu  $i(t)$ . Zeleně znázorněna výsledná efektivní hodnota.

Efektivní hodnota je vždy nezáporná a zpravidla větší nebo rovna střední hodnotě.

Z anglického překladu pro efektivní hodnotu „root mean square“ vyplývá, že jde o druhou odmocninu ze střední hodnoty kvadrátu původní funkce [3].

V případě periodických signálů, lze ve vztahu (2.1), (2.2), (2.4), (2.5) a (2.6) použít u integrálu i jiné meze. Například  $\langle -\frac{T_1}{2}, \frac{T_1}{2} \rangle$  nebo  $\langle t_k, t_k + T_1 \rangle$ , kde  $t_k$  je libovolný časový okamžik [1, 3].

Pokud je zkoumaný signál periodický, lze ho transformovat na Fourierovu řadu. Díky Fourierovým koeficientům  $c_k$  lze v některých případech vypočítat efektivní hodnotu signálu efektivněji [16]:

$$I_{\text{ef}} = \sqrt{\sum_{k=-\infty}^{\infty} |c_k|^2}, \quad (2.7)$$

při čemž platí:

$$|c_k| = \frac{A_k}{2}, \quad (2.8)$$

kde  $A_k$  je amplituda  $k$ -té harmonické funkce.

## 2.3 Poměrné činitele

Pro popis střídavých signálů se také využívají tzv. poměrné činitele [3]: činitel tvaru  $k_t$ , činitel výkyvu  $k_v$  a činitel plnění  $k_p$ . Jsou definovány následujícími vzorci [1, 3]:

$$k_t = \frac{I_{\text{ef}}}{I_s}, \quad (2.9)$$

$$k_v = \frac{I_m}{I_{\text{ef}}}, \quad (2.10)$$

$$k_p = \frac{I_s}{I_m}, \quad (2.11)$$

kde  $I_m$  je maximální hodnota průběhu, resp. amplituda u harmonického signálu. Index u veličiny efektivní hodnoty „ef“ se často vynechává, protože jde o nejčastěji používanou charakteristikou střídavých napětí a proudů. Místo  $I_{ef}$  se tak můžeme častěji setkat se zápisem  $I$  [3].

Činitel tvaru se využívá v levnějších měřicích přístrojích, kdy se efektivní hodnota získává cejchováním přístroje vynásobením činitele tvaru. Činitel výkyvu je hodnota, do které se zaručuje přesnost měřicího přístroje. Činitel plnění se používá především v elektroenergetice a silnoproudé elektrotechnice [1].

## 3 Teorie k apletu Diskrétní lineární a kruhová konvoluce v 1D

Konvoluce je matematická operace mezi dvěma signály. Značí se operátorem  $*$ . Může se provádět mezi analogovými i mezi diskrétními signály. Často se využívá pro výpočet odezvy na libovolný vstupní signál. U analogových systémů se hovoří o odezvě  $g(t)$ , u digitálních systémů o odezvě  $h[n]$ .

### 3.1 Lineární konvoluce

U spojitých signálů platí pro lineární konvoluci vztah [18]:

$$y(t) = g(t) * x(t) = \int_{-\infty}^{\infty} g(\tau)x(t - \tau)d\tau. \quad (3.1)$$

Konvoluce je operace, která má komutativní, asociativní a distributivní vlastnosti [25, 4]. Platí tedy:

$$a(t) * b(t) = b(t) * a(t), \quad (3.2)$$

$$a(t) * (b(t) * c(t)) = (a(t) * b(t)) * c(t), \quad (3.3)$$

$$a(t) * (b(t) + c(t)) = (a(t) * b(t)) + (a(t) * c(t)). \quad (3.4)$$

Příklad lineární konvoluce je na obrázku 3.1.

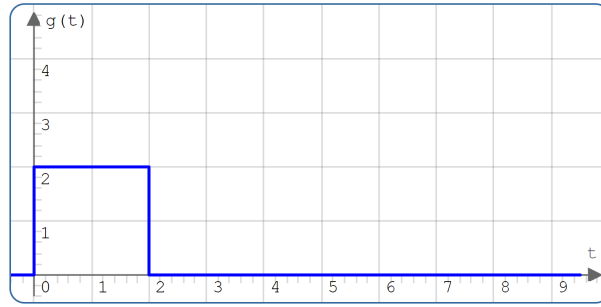
### 3.2 Diskrétní lineární konvoluce

Obecný vzorec pro diskrétní lineární konvoluci je [18]:

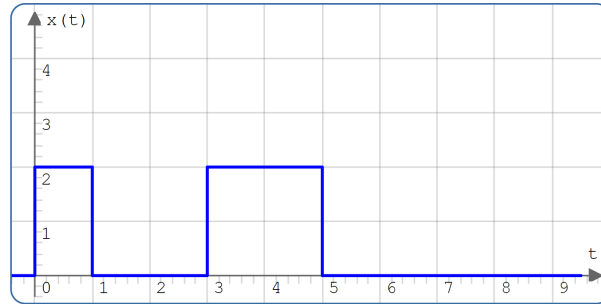
$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n - m]. \quad (3.5)$$

Pokud jsou známy délky signálů  $h[n]$  a  $x[n]$ , kde délka prvního signálu je označena jako  $H$  a délka druhého signálu jako  $X$ , lze snadno vypočítat výslednou délku konvoluce  $N = H + X - 1$ . Vztah pro diskrétní lineární konvoluci se tak zjednoduší: [18, 16]

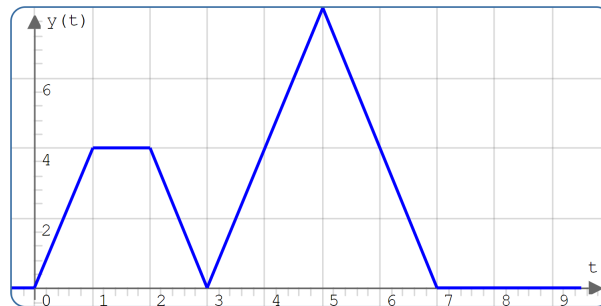
$$y[n] = h[n] * x[n] = \sum_{m=0}^{N-1} h[m]x[n - m]. \quad (3.6)$$



(a) Signál  $g(t)$ .



(b) Signál  $x(t)$ .



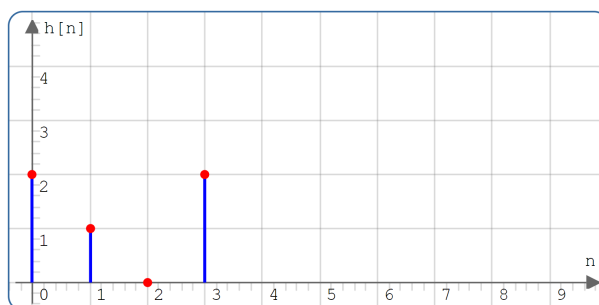
(c) Výsledek konvoluce  $y(t)$ .

Obr. 3.1: Příklad lineární konvoluce  $y(t) = g(t) * x(t)$ .

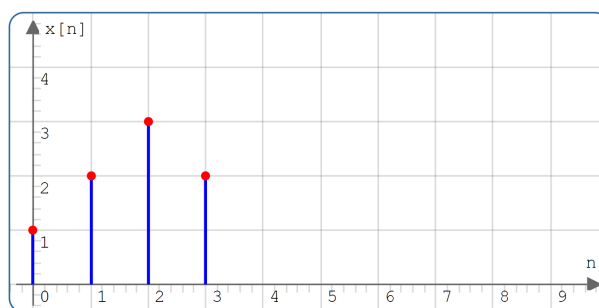
Příklad ručního výpočtu diskrétní lineární konvoluce zobrazen v tabulce 3.1. Příklad diskrétní lineární konvoluce je na obrázku 3.2.

Tab. 3.1: Příklad výpočtu diskrétní lineární konvoluce.

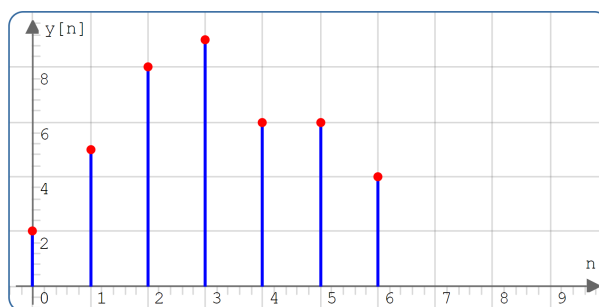
$m$	$n$	-3	-2	-1	0	1	2	3	4	5	6	7	$y[m]$
	$h[n]$				2	1	0	2					
	$x[n]$				1	2	3	2					
0	$x[-n]$	2	3	2	1								$1 \cdot 2 = 2$
1	$x[1-n]$		2	3	2	1							$2 \cdot 2 + 1 \cdot 1 = 5$
2	$x[2-n]$			2	3	2	1						$3 \cdot 2 + 2 \cdot 1 + 1 \cdot 0 = 8$
3	$x[3-n]$				2	3	2	1					$2 \cdot 2 + 3 \cdot 1 + 2 \cdot 0 + 1 \cdot 2 = 9$
4	$x[4-n]$					2	3	2	1				$2 \cdot 1 + 3 \cdot 0 + 2 \cdot 2 = 6$
5	$x[5-n]$						2	3	2	1			$2 \cdot 0 + 3 \cdot 2 = 6$
6	$x[6-n]$							2	3	2	1		$2 \cdot 2 = 4$
7	$x[7-n]$								2	3	2	1	0



(a) Signál  $h[n]$ .



(b) Signál  $x[n]$ .



(c) Výsledek diskrétní lineární konvoluce  $y[n]$ .

Obr. 3.2: Příklad diskrétní lineární konvoluce  $y[n] = h[n] * x[n]$ .

### 3.3 Diskrétní kruhová konvoluce

Při výpočtu diskrétní kruhové konvoluce se nejprve oba vstupní signály zperiodizují pomocí operace modulo  $N$  (viz rovnice (3.7)). Platí, že oba vstupní signály i výsledný signál mají stejnou délku  $N$ . Po provedení konvoluce mezi zperiodizovanými signály se tento výsledek vynásobí pravoúhlou okénkovou posloupností (viz rovnice (3.8)). Pro kruhovou konvoluci se užívá operátor  $\circledast$  [18, 21].

Periodický signál  $\tilde{h}[n]$  s periodou  $N$  vznikne zperiodizováním signálu  $h[n]$  s délkou  $N$  pomocí operace modulo [18]:

$$\tilde{h}[n] = h[\text{mod}_N(n)], \quad (3.7)$$

kde platí, že  $\text{mod}_N(n)$  udává zbytek po celočíselném dělení čísla  $n$  číslem  $N$ , který nabývá celočíselných hodnot z intervalu  $\langle 0; N - 1 \rangle$ .

Pravoúhlá okénková posloupnost je definovaná vztahem [18]:

$$R_N[n] = \begin{cases} 1, & \text{pro } n \in \langle 0; N - 1 \rangle, \\ 0, & \text{pro } n \notin \langle 0; N - 1 \rangle, \end{cases} \quad (3.8)$$

z čehož vyplývá, že se z kterýkoliv diskrétního signálu po vynásobení touto okénkovou posloupností zachová pouze prvních  $N$  vzorků (s indexy 0 až  $N - 1$ ) a ostatní vzorky budou nulové.

Obecný vztah pro diskrétní kruhovou konvoluci je [18]:

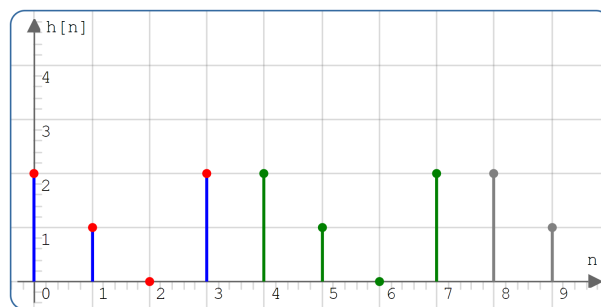
$$y[n] = h[n] \circledast x[n] = R_N[n] \sum_{m=-\infty}^{\infty} \tilde{h}[m] \tilde{x}[n - m], \quad (3.9)$$

Pokud jsou známy délky signálů  $h[n]$  a  $x[n]$ , kde délka prvního signálu je označena jako  $H$  a délka druhého signálu jako  $X$ , lze snadno vypočítat výslednou délku konvoluce  $N = H + X - 1$ . Jelikož u kruhové konvoluce platí, že délka obou signálů se rovná, výsledná délka konvoluce se může vypočítat také jako  $N = 2H - 1 = 2X - 1$ . Vztah pro diskrétní lineární konvoluci se tak zjednoduší: [18, 16, 21]

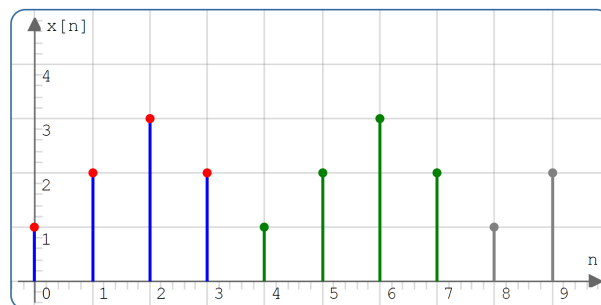
$$y[n] = h[n] \circledast x[n] = R_N[n] \sum_{m=0}^{N-1} \tilde{s}_1[m] \tilde{s}_2[n - m], \quad (3.10)$$

Příklad diskrétní kruhové konvoluce je na obrázku 3.3.

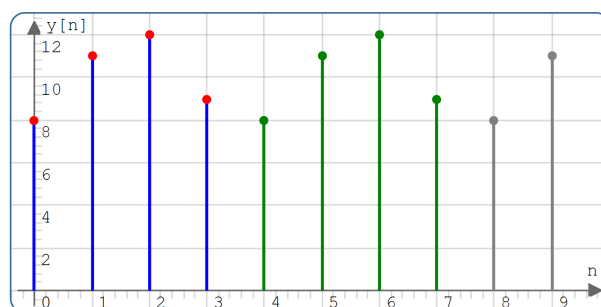




(a) Signál  $h[n]$ .



(b) Signál  $x[n]$ .



(c) Výsledek diskrétní kruhové konvoluce  $y[n]$ .

Obr. 3.3: Příklad diskrétní kruhové konvoluce  $y[n] = h[n] \otimes x[n]$ .

## 4 Teorie k apletu Diskrétní křížová korelace

Operace křížová korelace (podle [19] vzájemná korelace) udává míru podobnosti dvou signálů [18]. Některé zdroje ([19, 18]) označují výsledek korelace symbolem  $\gamma$ . V této práci je výsledek korelace označován jako  $y(t)$ , respektive  $y[n]$ , a operátor korelace jako  $\star$ .

Prakticky se křížové korelace využívá u radarů, kdy se vyslaný signál vrací odražený a zkreslený šumem. Pomocí korelace se vyhodnocuje podobnost obou signálů [25, 9].

### 4.1 Křížová korelace

U spojitých signálů platí pro křížovou korelaci vztah [15, 18]:

$$y(t) = g(t) \star x(t) = \int_{-\infty}^{\infty} g(\tau)x(\tau - t)d\tau. \quad (4.1)$$

V případě, kde si signály nebudou skoro vůbec podobné, korelace bude mít hodnotu blízkou nule. Maximální bude v případě, kdy první a druhý vstupní signál budou nejvíce podobné, respektive stejné. Pokud budou signály v protifázi, tedy budou mít stejný tvar, ale s opačným znaménkem, bude výsledná funkce minimální [25, 8].

Dále je definována autokorelační funkce, pro kterou platí, že oba vstupní signály jsou totožné [16]. Tedy  $g(t) = x(t)$ .

Příklad křížové korelace je na obrázku 4.1.

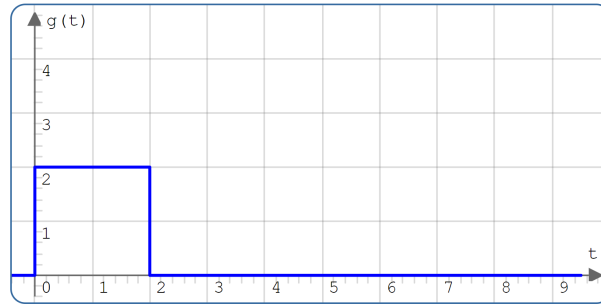
### 4.2 Diskrétní křížová korelace

V případě diskrétních signálů lze křížovou korelaci vypočítat vztahem [19, 15]:

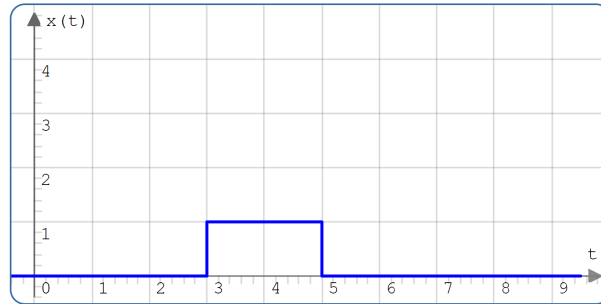
$$y[n] = h[n] \star x[n] = \sum_{m=-\infty}^{\infty} h[m]x[m - n]. \quad (4.2)$$

Pokud jsou známy délky signálů  $h[n]$  a  $x[n]$ , kde délka prvního signálu je označena jako  $H$  a délka druhého signálu jako  $X$ , lze snadno vypočítat výslednou délku konvoluce  $N = H + X - 1$ . Vztah pro diskrétní lineární konvoluci se tak zjednoduší: [18, 16]

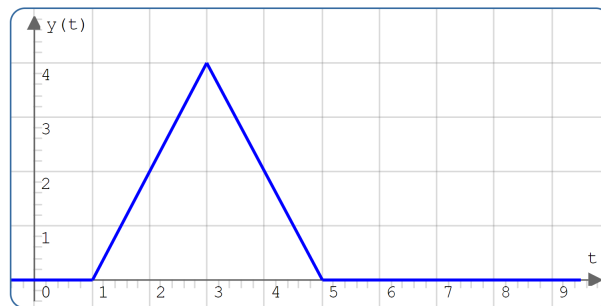
$$y[n] = h[n] \star x[n] = \sum_{m=0}^{N-1} h[m]x[m - n]. \quad (4.3)$$



(a) Signál  $g(t)$ .



(b) Signál  $x(t)$ .



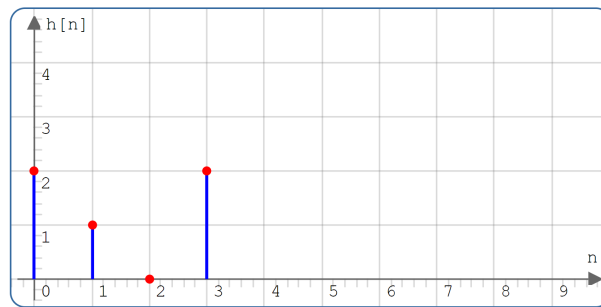
(c) Výsledek korelace  $y(t)$ .

Obr. 4.1: Příklad křížové korelace  $y(t) = g(t) \star x(t)$ .

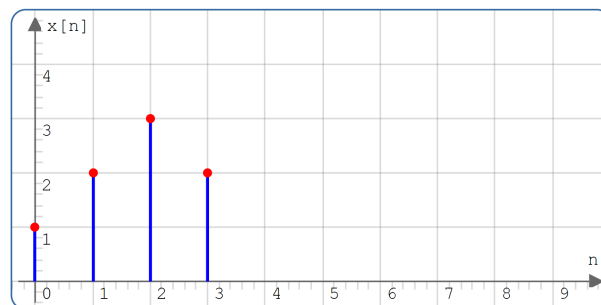
Korelace se podobá konvoluci. Z následujícího vztahu vyplývá, že konvoluce odpovídá korelaci signálu  $h$  a  $x$ , který je ovšem časově obrácený [10]:

$$h[n] \star x[n] = h[n] * x[-n]. \quad (4.4)$$

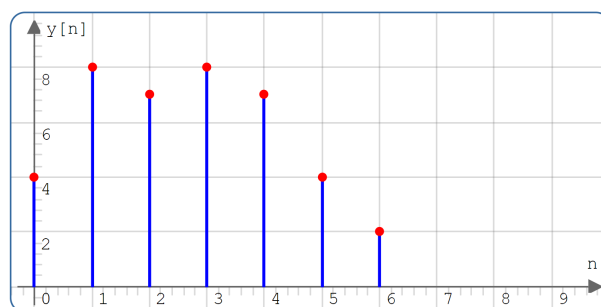
Příklad diskrétní křížové korelace je na obrázku 4.2.



(a) Signál  $h[n]$ .



(b) Signál  $x[n]$ .



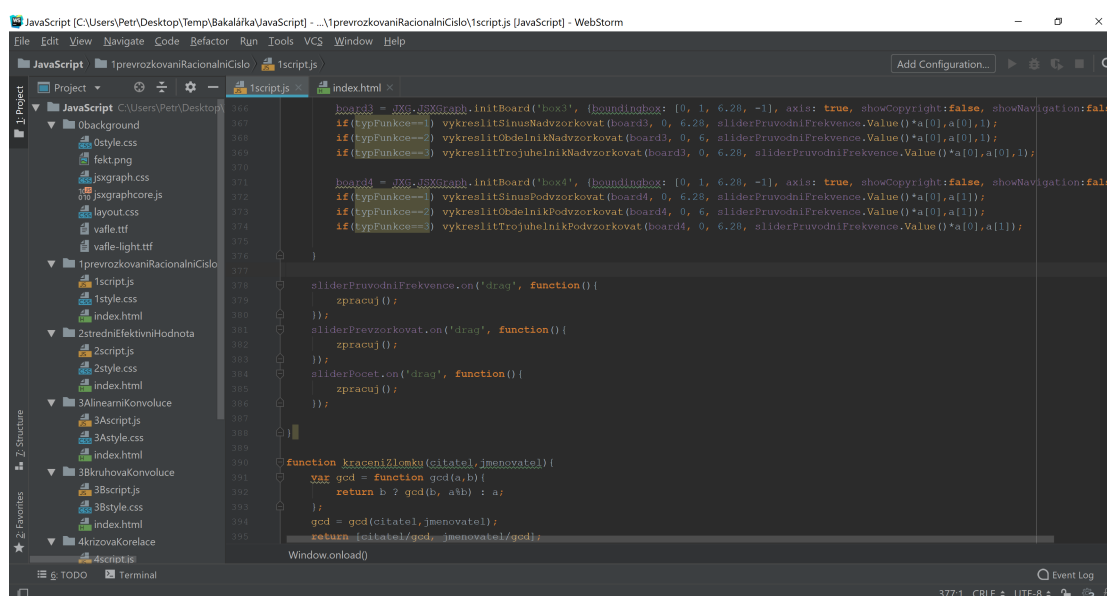
(c) Výsledek diskrétní korelace  $y[n]$ .

Obr. 4.2: Příklad diskrétní křížové korelace  $y[n] = h[n] \star x[n]$ .

## 5 Vývoj apletů

V následující kapitole je popsáno, které programovací jazyky byly použity při vytváření apletů. Dále je zde uvedeno, jak byly jednotlivé aplety testovány.

Při vytváření všech apletů bylo použito vývojové prostředí WebStorm od společnosti JetBrains s.r.o. ve studentské licenci. Grafické rozhraní vývojového prostředí je na obrázku 5.1.



Obr. 5.1: Vývojové prostředí WebStorm.

### 5.1 Použité jazyky

Při vytváření jednotlivých aplikací byl použit programovací jazyk JavaScript. Grafický vzhled aplikací je vytvořen pomocí kaskádových stylů CSS. Všechny komponenty jsou vloženy pomocí jazyka HTML do webové stránky. V této části jsou jednotlivé jazyky stručně popsány.

#### 5.1.1 JavaScript

JavaScript je programovací, resp. skriptovací jazyk, který vyvinula v roce 1995 firma Netscape. Jedná se o objektově orientovaný jazyk, který se používá ve stránkách HTML. Syntaxe jazyka vychází z jazyka C++. Je „case sensitive“, rozlišuje tedy velká a malá písmena. Skripty jsou interpretovány přímo ve webovém prohlížeči, není třeba kompilátor. Jeho výhodou je, že se na všech operačních systémech a webových prohlížečích chová totožně [13].

Pomocí tohoto programovacího jazyka lze psát skripty, které doplňují možnosti HTML. Webové stránky se tak mohou stát dynamickými. Zdrojový text JavaScriptu se píše nebo vkládá přímo do jazyka HTML. Pokud je skript uložený v externím souboru, má koncovku `.js` [24].

Základní možnosti JavaScriptu jsou: reakce na události (např. pohyb myši, kliknutí, stisknutí klávesy a další), práce s formuláři včetně ověření vstupních dat, měnit obsah stránky v závislosti na datu nebo hodině [14].

JavaScript je odlišný jazyk než jazyk Java, která je vytvořena firmou Sun Microsystems, a navíc je to jazyk kompilovaný. Java je schopná tvořit i složitější programy, které se vykonávají rychleji. Její nevýhoda je však v tom, že v posledních letech webové prohlížeče přestávají podporovat Java applety – programy určené pro webové prostředí. Je to z toho důvodu, že těmto Java appletům přisuzují webové prohlížeče bezpečnostní hrozbu [17].

Příklad kódu v jazyce JavaScript je ve výpisu 5.1.

### Knihovna JSXGraph

Pro interaktivní ovládání apletů byla využita knihovna JSXGraph ve svobodné licenci. Obsáhlá dokumentace ke knihovně je volně dostupná na webových stránkách <https://jsxgraph.org/>.

Díky použité knihovně lze do webového prostředí vykreslit grafy a do nich jednoduše přidávat následující komponenty: osy, popisky, body, přímky, funkce, geometrické tvary, Bézierovy křivky a další. Každý tento objekt může mít své definované vlastnosti: barvu, velikost, tvar, pohyblivost v grafu a další.

Kromě grafů knihovna umí knihovna prací s posuvníky (slidery), které ve vytvořených apletech slouží jako hlavní interaktivní prvky. Každému prvku lze totiž přidělit událost, která při manipulaci s prvkem vykonává požadované funkce.

Výpis 5.1: Příklad použití jazyka JavaScript.

```
1 board = JXG.JSXGraph.initBoard('box',
2     {boundingbox: [-5, 5, 5, -5], axis: true});
3
4 efVysledek = Math.sqrt(1 / periodaObdelnik * obsahObdelnik2);
5 sVysledek = 1 / periodaObdelnik * obsahObdelnik;
6
7 stredni.innerHTML = "I<sub>s</sub>=" + zaokr(sVysledek);
8 efektivni.innerHTML = "I<sub>ef</sub>=" + zaokr(efVysledek);
```

## 5.1.2 HTML

Jazyk HTML (Hypertext Markup Language) je nerozšířenější značkovací jazyk, který slouží pro vytváření webových stránek. Vykreslení souboru napsaném v jazyce HTML zajišťují webové prohlížeče [2].

Pomocí jazyka HTML lze vkládat do webové stránky základní komponenty, jako jsou například tabulky, odstavce textu, obrázky, prvky formuláře, odkazy a další. Koncovka souboru, ve kterém je kód HTML uložen, je `.html` [6].

Jednotlivé příkazy jsou vkládány mezi značky, tzv. tagy, které jsou z velké části párové. Každý tag může mít několik tzv. atributů, které ke značce přiřazuje důležité vlastnosti [12].

Vznik jazyka HTML je datován k roku 1990. V současné době (2019) je aktuální verze HTML 5.0 [2].

Ve výpisu 5.2 je zobrazen příklad kódu jazyka HTML.

Výpis 5.2: Příklad použití jazyka HTML.

```
1 <h2>Vyberte typ funkce:</h2>
2
3 <form name="form">
4     <select name="typFunkce"; onchange=aktualizovat(value);>
5         <option value=1>sinus</option>
6         <option value=2>obdélník</option>
7         <option value=3>trojúhelník</option>
8     </select>
9 </form>
```

## 5.1.3 CSS

Kaskádové styly zajišťují grafickou podobu webové stránky. Umí u každého prvku HTML definovat, jak se vykreslí ve webovém prohlížeči. Základními parametry, které lze nastavovat, jsou barva textu a pozadí, délka a šířka prvků, formátování a fonty textů. Koncovka souboru, ve kterém je kód CSS uložen, je `.css` [22].

Hlavní výhodou jazyka CSS je, že lze díky němu nastavit grafickou podobu jednotlivých skupin příkazů HTML. Například tedy uživatel jednou, jak má vypadat nadpis první úrovně. Těchto nadpisů lze v HTML kódu použít libovolně mnoho. Pokud bude uživatel chtít podobu nadpisů změnit, nemusí toto formátování provádět u každého nadpisu zvlášť, ale postačí pouze na jednom místě – v kódu CSS [22].

Výpis 5.3 ukazuje možné použití kaskádových stylů.

Výpis 5.3: Příklad použití jazyka CSS.

```

1 @font-face {
2     font-family: vafle;
3     src: url("../0background/vafle.ttf");
4 }
5
6 #box1, #box2 {
7     width:400px;
8     float:left;
9 }

```

## 5.2 Testování

Základním požadavkem pro bezchybné spuštění apletů je, aby zvolený webový prohlížeč povoloval JavaScript. Pokud prohlížeč JavaScript nepodporuje, nebo ho uživatel ve vnitřním nastavení prohlížeče zakáže, aplety nebudou funkční v plné míře. V případě, že JavaScript není povolen, v místě všech grafů se vypíše hláška: „Pokud se na tomto místě nevykresluje graf, JavaScript nefunguje.“ Viz obrázek 5.2. S podporou jazyků HTML a CSS zpravidla nebývá problém, protože bývají součástí téměř každého webu.

**FEKT Převzorkování diskrétního signálu v poměru racionálního čísla**

**Vyberte typ funkce:**  
sinus

**Zadejte původní vzorkovací frekvenci, čísel L pro nadvzorkování a čísel D pro podvzorkování:**

Pokud se na tomto místě nevykresluje graf, JavaScript nefunguje.

$f_{vzn} = \frac{1}{10} \cdot f_{vz}$   
 $f_{vzn} = \frac{10}{15} \cdot f_{vz}$   
 $f_{vzn} = \frac{2}{3} \cdot f_{vz}$

1.) Původní funkce:  
Pokud se na tomto místě nevykresluje graf, JavaScript nefunguje.

2.) Přidání nulových vzorků:  
Pokud se na tomto místě nevykresluje graf, JavaScript nefunguje.

3.) Nadvzorkování po filtraci:  
Pokud se na tomto místě nevykresluje graf, JavaScript nefunguje.

4.) Výsledná převzorkovaná funkce:  
Pokud se na tomto místě nevykresluje graf, JavaScript nefunguje.

Obr. 5.2: Ukázka chování apletu, pokud není povolen JavaScript.

Při vývoji jednotlivých apletů byl používán převážně prohlížeč Google Chrome (verze 74.0.3729.157). Dále byly aplikace funkční v prohlížeči Microsoft Edge (verze 42.17134.1.0). Testováno na notebooku HP Envy 13 (operační systém Windows 10).



S problémy se lze setkat u prohlížeče Internet Explorer (testováno ve verzi 11.706.17134.0), který dokázal při testování spustit pouze dva aplety (Převzorkování diskrétního signálu v poměru racionálního čísla, Střední a efektivní hodnota signálu). V ostatních zmiňovaných internetových prohlížečích pracovaly všechny aplety správně.

V těchto prohlížečích byly zdrojové soubory načítány offline z disku počítače i z online verze, která byla z důvodu testování dočasně dostupná na webových stránkách <http://signal.y9e.cz/>.

Všechny aplety jsou uzpůsobeny i pro bezproblémové spuštění na mobilním telefonu. Protože dotykový mobilní telefon nereaguje na JavaScriptovou událost `mouseup`, musely být stejné funkce duplikovány pro událost `touchend`.

Aplikace jsou uživatelsky přívětivější v režimu telefonu na šířku. V režimu na výšku je sekce se vstupní a výstupní částí apletů zmenšena a čitelnost a ovládání apletů je pro uživatele těžší.

Aplety byly v online verzi testovány na telefonu Samsung Galaxy A50 (operační systém Android 9.0 Pie) v prohlížečích Chrome (verze 74.0.3729.157) a Samsung Internet (verze 9.2.00.70). Prohlížeč Samsung Internet překlopil aplety do nočního režimu, kde se ze světlých ploch a textů staly tmavé a naopak. Na funkcionalitu apletů to však nemělo žádný vliv.

Dále byly aplety testovány na zařízení iPhone 6S Plus (operační systém iOS 12.3) v prohlížeči Safari (verze 12.0.3) s bezproblémovým výsledkem.

## 6 Popis vytvořených apletů

V této kapitole je uveden popis jednotlivých vytvořených apletů. Všechny aplikace lze spustit v libovolném webovém prohlížeči, který podporuje JavaScript. V některých případech se webový prohlížeč může uživatele dotázat, zda má JavaScript povolit. To je nutné pro funkčnost všech aplikací.

Všechny aplety mají podobný vizuální vzhled, který má působit moderním dojmem. Ve vrchní části apletu se vždy nachází hlavička s logem Fakulty elektrotechniky a komunikačních technologií, které slouží také jako odkaz na rozcestník, kde se lze přepínat mezi vytvořenými aplikacemi. Dále je v hlavičce uveden název apletu.

Hlavní část programu je rozdělena na vstupní a výstupní část. Ve vstupní části, která se nachází vlevo, se zpravidla volí signál, který má být zpracován. V pravé výstupní části se zpravidla vykreslují grafy s požadovanými výsledky.

Ve spodní části je uveden popis každého apletu a u většiny apletů i zjednodušený postup požadované operace. V patičce se nachází informace o roku vzniku apletu, jméno autora a zkratka fakulty, pro kterou byly programy vyvinuty.

### 6.1 Převzorkování diskrétního signálu v poměru racionálního čísla

Na obrázku 6.1 je zobrazena grafická podoba apletu Převzorkování diskrétního signálu v poměru racionálního čísla.

#### Vstupní část apletu

V levé vstupní části aplikace může uživatel typ funkce, která má být převzorkována. V rozbalovací nabídce může vybrat jednu ze časově spojitých funkcí: harmonický signál sinus, obdélníkový signál, nebo trojúhelníkový signál.

Dále lze pomocí tří sliderů ovlivňovat parametry pro převzorkování signálu v poměru racionálního čísla. První sliderem se volí původní vzorkovací kmitočet, kterým je jednoduše navzorkován zadaný vstupní signál. Druhým sliderem lze měnit činitel  $L$  pro nadvzorkování původního navzorkovaného signálu. Třetí slider ovládá činitel  $D$  pro podvzorkování již nadvzorkovaného signálu.

Pod polem se slidery jsou vypsány rovnice, pomocí kterých lze ze zadaných hodnot (původního vzorkovacího kmitočtu, činitele  $L$  pro nadvzorkování a činitele  $D$  pro podvzorkování) vypočítat novou hodnotu vzorkovací frekvence po převzorkování. Pokud jsou zadaná čísla  $L$  a  $D$  soudělná, zlomek  $\frac{L}{D}$  se vykrátí a dále se počítá s aktualizovanými hodnotami  $L$  a  $D$ . Tím se dosáhne stejného výsledku, avšak s několikanásobně jednodušším procesem.

## Výstupní část apletu

V pravé výstupní části jsou umístěny čtyři očíslované grafy, ve kterých je postupně znázorněn proces převzorkování diskrétního signálu v poměru racionálního čísla.

První graf zobrazuje zvolenou časově spojitou funkci, která je obálkou pro navzorkování podle zadané hodnoty. Pro zvolenou funkci sinus se zobrazuje jedna perioda, pro obdélníkovou funkci jedna perioda skládající se ze čtyř různých podfunkcí, u trojúhelníkové funkce se zobrazuje jedna perioda skládající se z jednoho trojúhelníku nad časovou osou a jednoho trojúhelníku pod časovou osou. Spolu s touto obálkou je zobrazen navzorkovaný signál, který má zvolenou původní vzorkovací frekvenci  $f_{vz}$ .

Druhý graf znázorňuje první krok nadvzorkování, tedy přidání nulových vzorků. Mezi každý z původních vzorků je rovnoměrně přidáno  $L - 1$  nulových vzorků. V grafu jsou zvýrazněny zelenou barvou. Aktuální frekvence signálu je  $L \cdot f_{vz}$ .

Třetí graf znázorňuje, jak se změní signál po filtraci. Po nadvzorkování – přidání nulových vzorků – probíhá filtrace dolní propustí, díky které se nulové vzorky aproximují podle zvoleného typu funkce. Tyto aproximované vzorky jsou v grafu znázorněny zeleně. Před následným podvzorkováním opět filtruje dolní propustí, aby nedošlo k aliasingu. V procesu převzorkování se vyskytují dvě filtrace dolní propustí bezprostředně za sebou. Lze je tedy sloužit do jednoho filtru typu dolní propust. Aktuální frekvence signálu zůstává  $L \cdot f_{vz}$ .

Poslední čtvrtý graf zobrazuje signál po podvzorkování podle činitele  $D$ . Podvzorkovává se předchozí již nadvzorkovaný signál. Po každém zachovaném vzorku se  $D - 1$  vzorků vynuluje, respektive odebere. Graf tedy zobrazuje výsledný převzorkovaný signál. Konečný signál má kmitočet  $\frac{L}{D} \cdot f_{vz}$ .

Vyberte typ funkce:

sinus ▼

Zadejte původní vzorkovací frekvenci,  
činitel L pro nadvzorkování a činitel D  
pro podvzorkování:

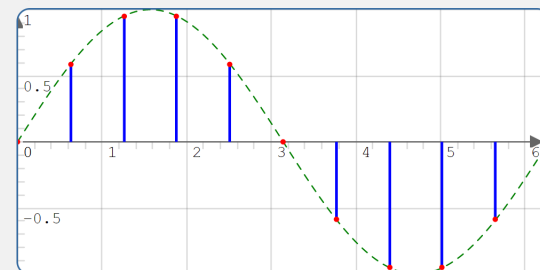


$$f_{vz_n} = L/D \cdot f_{vz}$$

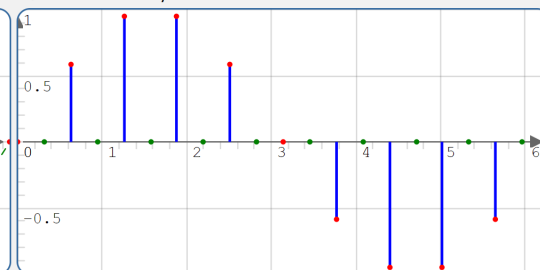
$$f_{vz_n} = 10/15 \cdot f_{vz}$$

$$f_{vz_n} = 2/3 \cdot f_{vz}$$

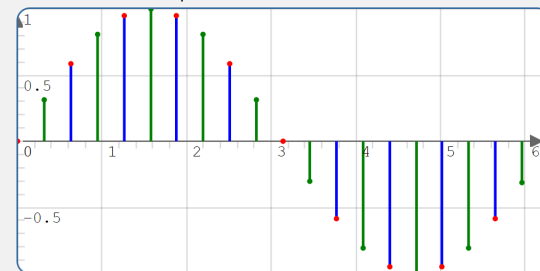
1.) Původní funkce:



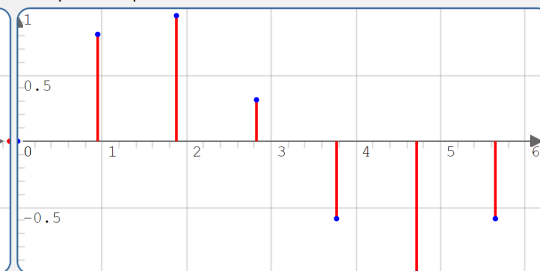
2.) Přidání nulových vzorků:



3.) Nadvzorkování po filtraci:



4.) Výsledná převzorkovaná funkce:



Obr. 6.1: Aplet Převzorkování diskretního signálu v poměru racionálního čísla.

## 6.2 Střední a efektivní hodnota signálu

Aplikace pro vypočtení střední a efektivní hodnoty signálu je zobrazena na obrázku 6.2. Program je rozdělen do dvou částí – vstupní nalevo a výstupní napravo.

### Vstupní část apletu

Ve vstupní části si uživatel vybírá signál, který je zkoumán. Na výběr má ze tří základních možností: harmonický signál sinus, obdélníkový signál nebo trojúhelníkový signál.

Pod volbou funkce je zobrazeno pole pro úpravu parametrů zvolené funkce. U funkce sinus lze pomocí tří sliderů ovlivňovat její koeficienty. První slider ovládá amplitudu zvolené funkce, pomocí druhého slideru lze měnit úhlovou frekvenci harmonické funkce a poslední třetí slider ovládá fázový posun sinusoidy.

Obdélníkový, respektive schodovitý, signál lze ovlivňovat pomocí dvojice polí s posuvníky. První slider určuje, kolik schodů bude mít zkoumaný signál. Při čemž každý schod má jednotkovou velikost. Podle tohoto parametru se ve druhém poli s posuvníky zobrazí požadovaný počet sliderů. Každý z těchto posuvníků ovládá hodnotu funkce v daném úseku.

U trojúhelníkového signálu lze pomocí prvního slideru měnit směrnici přímky, která tvoří první polovinu rovnoramenného trojúhelníku. Trojúhelníkový signál tak bude symetrický – sestupná strana trojúhelníku bude stejně velkou směrnici s opačným znaménkem. Druhý slider udává, na jak velkém intervalu bude vykreslena první polovina trojúhelníku. Celkový interval zkoumaného trojúhelníkového signálu bude tedy dvojnásobný.

Pod polem s nastavování parametrů se dle vybraného typu signálu v případě sinusoidy nebo trojúhelníkového signálu zobrazí matematický popis zadané funkce. V případě obdélníkového signálu lze popis funkce jednoduše vyčíst z pole zadaných parametrů.

Ve spodní části se zobrazují vypočítaná střední a efektivní hodnota zvoleného signálu. Jednotlivé hodnoty jsou barevně rozlišeny kvůli snadnější orientaci ve výstupním grafu, kde jsou tyto hodnoty vykresleny.

### Výstupní část apletu

Ve výstupní části programu se nachází graf, ve kterém je vykreslen časový průběh zadaného signálu. Na základě vstupních hodnot se vypočte střední a efektivní hodnota a vykreslí se do grafu. Střední hodnota je zakreslena červenou přerušovanou čarou a efektivní hodnota zelenou přerušovanou čarou.

Vyberte typ funkce:

obdélník ▾

Zadejte parametry:

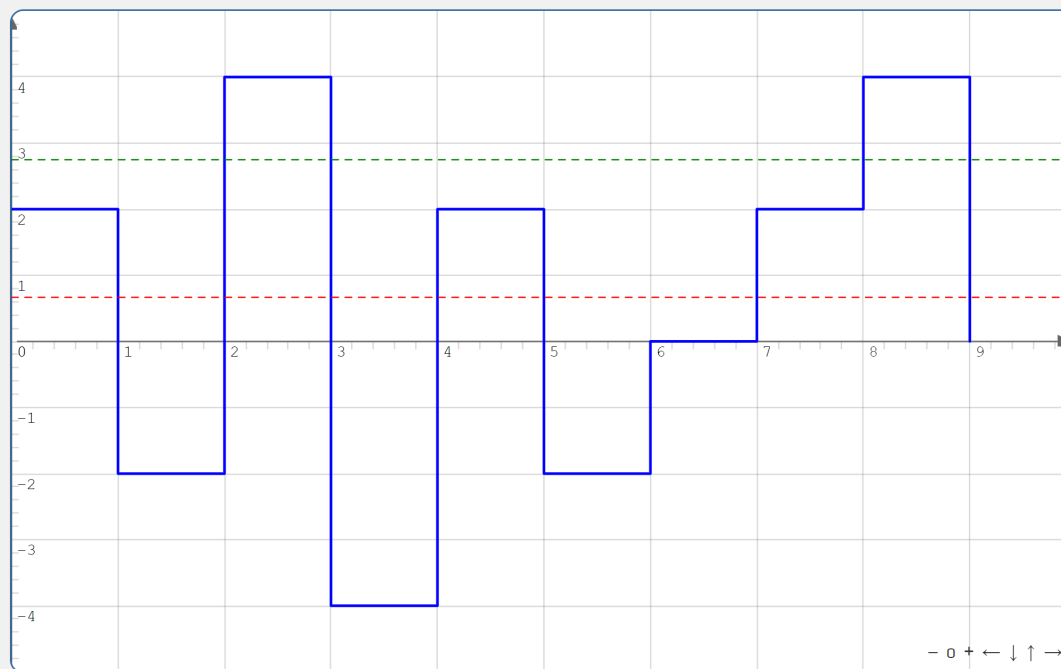
Pocet schodu = 9.00

- 1. schod = 2.00
- 2. schod = -2.00
- 3. schod = 4.00
- 4. schod = -4.00
- 5. schod = 2.00
- 6. schod = -2.00
- 7. schod = 0.00
- 8. schod = 2.00
- 9. schod = 4.00

Vypočtené hodnoty:

$I_s = 0.667$

$I_{ef} = 2.749$



Obr. 6.2: Aplet Střední a efektivní hodnota signálu.

## 6.3 Diskrétní lineární a kruhová konvoluce v 1D

V následující kapitole jsou popisovány dva aplety. Jeden pro výpočet diskrétní lineární konvoluce v 1D a druhý pro výpočet diskrétní kruhové konvoluce v 1D. Hlavní rozdíl v aplikacích je především v tom, že druhý zmíněný aplet pracuje s druhým vstupním signálem jako se zperiodizovaným.

### 6.3.1 Diskrétní lineární konvoluce v 1D

Grafická podoba aplikace pro výpočet diskrétní lineární konvoluce v 1D je zobrazena na obrázku 6.3.

#### Vstupní část apletu

V levé části apletu jsou dva interaktivní grafy. Každý slouží pro zadávání jednoho ze dvou vstupních signálů, mezi kterými bude probíhat operace lineární konvoluce. Signál lze upravit jednoduchým potažením červeného bodu na požadovanou hodnotu. Každý bod lze přetáhnout na libovolnou pozici v grafu. Index (hodnota na horizontální ose) přetahovaného bodu se však po stisknutí tlačítka myši nezmění, změní se pouze jeho hodnota (na vertikální ose), jak by se dalo očekávat. Pokud má požadovaný signál mít menší počet vzorků, než je ve výchozím nastavení, stačí nadbytečné prvky vynulovat – nastavit jim nulovou hodnotu.

Pod každým grafem se nacházejí dvě tlačítka, které obsluhují jednotlivé grafy. Stisknutí tlačítka **Vynulovat** způsobí, že všechny prvky v požadovaném grafu budou mít nulovou hodnotu. Tlačítko **Náhodně** slouží k zadání nových náhodných hodnot zvoleného signálu. I tyto vynulované nebo náhodně zvolené prvky lze snadno potažením myši upravit.

#### Výstupní část apletu

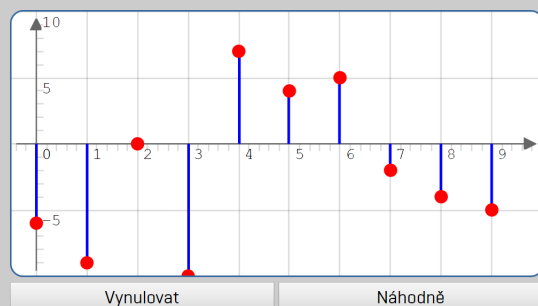
V pravé části apletu se nachází čtyři očíslované bloky. V prvním bloku je beze změny zkopírovaný první zadáný signál. V druhém bloku je graf s časově obráceným druhým signálem. Časového obrácení lze dosáhnout znegováním všech čísel indexů. Ve výchozím zobrazení je tento druhý časově obrácený signál posunut o 20 prvků v kladném směru (ke všem znegovaným indexům bylo přičteno číslo 20).

Ve třetím bloku se nachází slider, který ovlivňuje posunutí druhého časově obráceného signálu, zobrazeného ve druhém grafu ve výstupní části apletu. Tímto posouváním zmiňovaného signálu směrem od mínus nekonečna do nekonečna lze demonstrovat postup, kterým se počítá lineární konvoluce. Ve výchozím stavu má slider hodnotu posunutí 20, aby byl v následujícím bloku zobrazen konečný výsledek lineární konvoluce.

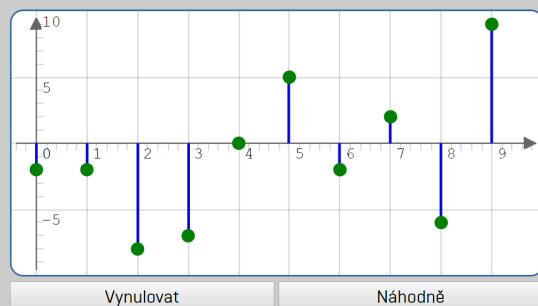
V posledním bloku se nachází graf s (částečným) výsledkem lineární konvoluce mezi prvním a druhým signálem. Částečnost vykresleného výsledku závisí na hodnotě posunutí druhého časově obráceného signálu. Pokud je slider posunutí v záporných číslech, není vykreslen žádný výsledek. První vzorek se vykresluje, při hodnotě slideru 0 – oba vstupní signály se časově překrývají právě v jednom bodě. První vykreslený prvek (s indexem 0) ve výstupním grafu se spočítá jako součin prvků prvního a druhého signálu, které se překrývají (mají totožné indexy). Každý další následující prvek se počítá jako součet součinů všech prvků prvního a druhého signálu, které se překrývají (mají shodné indexy). Pokud slider posunutí přesáhne hodnotu 19, výsledný graf zobrazuje úplný výsledek lineární konvoluce signálů.



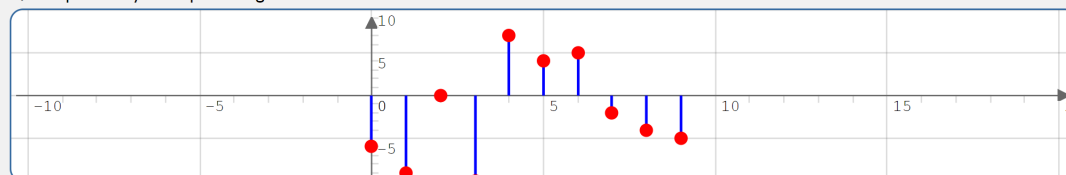
Zadejte první signál:



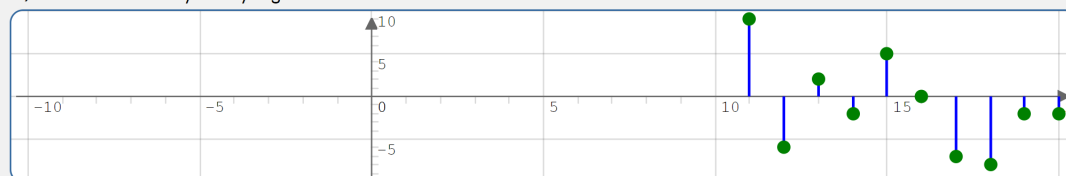
Zadejte druhý signál:



1.) Zkopírovaný fixní první signál:



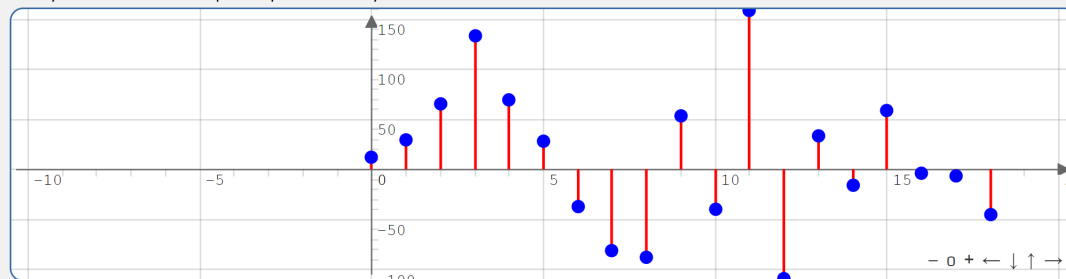
2.) Časově obrácený druhý signál:



3.) Posunutí druhého obráceného signálu o 20:



4.) Výsledek konvoluce podle posunutí (vyskreseno 19 z 19):



Obr. 6.3: Aplet Diskrétní lineární konvoluce v 1D.

### 6.3.2 Diskrétní kruhová konvoluce v 1D

Na obrázku 6.4 se nachází grafické prostředí apletu pro výpočet diskrétní kruhové konvoluce v 1D.

#### Vstupní část apletu

V levé části apletu jsou dva interaktivní grafy. Každý slouží pro zadávání jednoho ze dvou vstupních signálů, mezi kterými bude probíhat operace kruhová konvoluce. Signál lze upravit jednoduchým potažením červeného bodu na požadovanou hodnotu. Každý bod lze přetáhnout na libovolnou pozici v grafu. Index (hodnota na horizontální ose) přetahovaného bodu se však po stisknutí tlačítka myši nezmění, změní se pouze jeho hodnota (na vertikální ose), jak by se dalo očekávat. Oba dva vstupní signály, stejně jako výstupní signál, mají z podstaty kruhové konvoluce stejný počet prvků.

Pod každým grafem se nacházejí dvě tlačítka, které obsluhují jednotlivé grafy. Stisknutí tlačítka **Vynulovat** způsobí, že všechny prvky v požadovaném grafu budou mít nulovou hodnotu. Tlačítko **Náhodně** slouží k zadání nových náhodných hodnot zvoleného signálu. I tyto vynulované nebo náhodně zvolené prvky lze snadno potažením myši upravit.

#### Výstupní část apletu

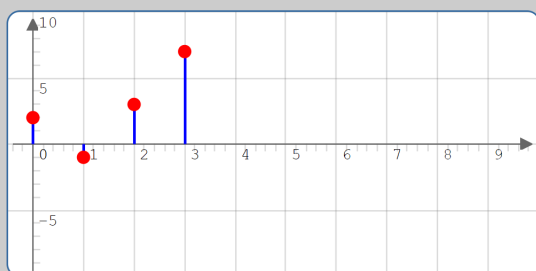
V pravé části apletu se nachází čtyři očíslované bloky. V prvním bloku je zkopírovaný první zadaný signál, který je pro lepší představu zperiodizován. Zadaná perioda se oproti ostatním hodnotám zobrazuje zvýrazněně. V druhém bloku je graf s časově obráceným druhým signálem, který je zperiodizován. Zadaná perioda je v grafu zobrazena červeně, sousední periody jsou v grafu znázorněny zeleně, ostatní periody jsou vykresleny světle šedou barvou.

Ve třetím bloku se nachází slider, který ovlivňuje posunutí druhého časově obráceného zperiodizovaného signálu, zobrazeného ve druhém grafu ve výstupní části apletu. Tímto posouváním zmiňovaného signálu směrem od mínus nekonečna do nekonečna lze demonstrovat postup, kterým se počítá kruhové konvoluce. Ve výchozím stavu má slider hodnotu posunutí 3, aby byl v následujícím bloku zobrazen konečný výsledek kruhové konvoluce.

V posledním bloku se nachází graf s (částečným) výsledkem kruhové konvoluce mezi prvním a druhým signálem. Částečnost vykresleného výsledku závisí na hodnotě posunutí druhého časově obráceného zperiodizovaného signálu. Pokud je slider posunutí v záporných číslech, není vykreslen žádný výsledek. První vzorek se vykresluje, při hodnotě slideru 0 – pozice slideru se nachází nad prvním prvkem (s indexem 0) prvního signálu. První vykreslený prvek (s indexem 0) ve výstupním

grafu se spočítá jako součet součinů prvků prvního a druhého signálu, které se překrývají (mají totožné indexy). Každý další následující prvek se počítá shodně jako součet součinů všech prvků prvního a druhého signálu, které se překrývají (mají shodné indexy). Pokud slider posunutí přesáhne hodnotu 3 (přesáhne délku prvního signálu), výsledný graf zobrazuje úplný výsledek kruhové konvoluce signálů.

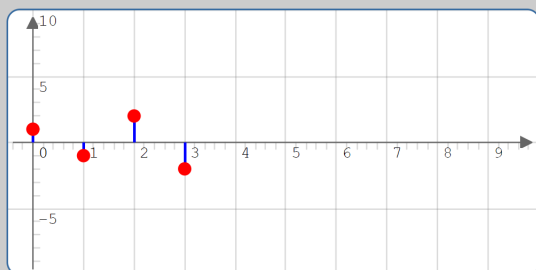
Zadejte jednu periodu prvního signálu:



Vynulovat

Náhodně

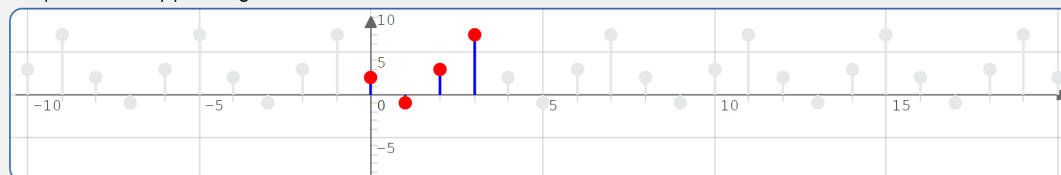
Zadejte jednu periodu druhého signálu:



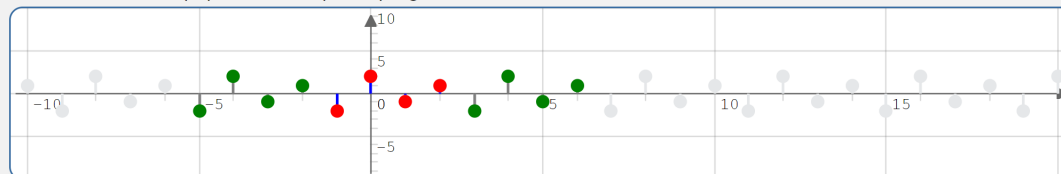
Vynulovat

Náhodně

1.) Zperiodizovaný první signál:



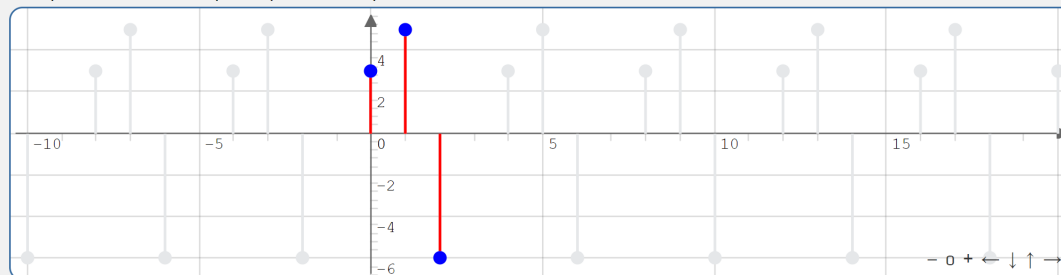
2.) Časově obrácený zperiodizovaný druhý signál:



3.) Posunutí druhého obráceného zperiodizovaného signálu o 2:



4.) Výsledek konvoluce podle posunutí (vyskreseno 3 z 4):



Obr. 6.4: Aplet Diskrétní kruhová konvoluce v 1D.

## 6.4 Diskrétní křížová korelace

Na obrázku 6.5 je zobrazen aplet pro výpočet diskrétní křížové korelace.

### Vstupní část apletu

V levé části apletu jsou dva interaktivní grafy. Každý slouží pro zadávání jednoho ze dvou vstupních signálů, mezi kterými bude probíhat operace křížové korelace. Signál lze upravit jednoduchým potažením červeného bodu na požadovanou hodnotu. Každý bod lze přetáhnout na libovolnou pozici v grafu. Index (hodnota na horizontální ose) přetahovaného bodu se však po stisknutí tlačítka myši nezmění, změní se pouze jeho hodnota (na vertikální ose), jak by se dalo očekávat. Pokud má požadovaný signál mít menší počet vzorků, než je ve výchozím nastavení, stačí nadbytečné prvky vynulovat – nastavit jim nulovou hodnotu.

Pod každým grafem se nacházejí dvě tlačítka, které obsluhují jednotlivé grafy. Stisknutí tlačítka **Vynulovat** způsobí, že všechny prvky v požadovaném grafu budou mít nulovou hodnotu. Tlačítko **Náhodně** slouží k zadání nových náhodných hodnot zvoleného signálu. I tyto vynulované nebo náhodně zvolené prvky lze snadno potažením myši upravit.

### Výstupní část apletu

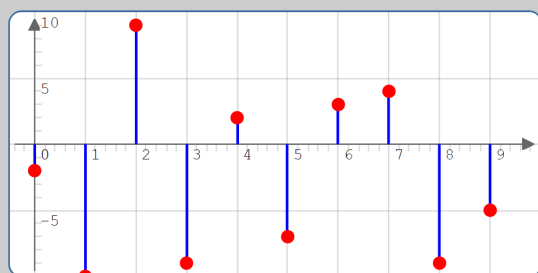
V pravé části apletu se nachází čtyři očíslované bloky. V prvním bloku je beze změny zkopírovaný první zadaný signál. V druhém bloku je zkopírovaný druhý zadaný signál, který však může být časově posunutý. Ve výchozím zobrazení je tento druhý signál posunut o 20 prvků v kladném směru (ke všem znegovaným indexům bylo přičteno číslo 20).

Ve třetím bloku se nachází slider, který ovlivňuje posunutí druhého signálu, zobrazeného ve druhém grafu ve výstupní části apletu. Tímto posouváním zmiňovaného signálu směrem od mínus nekonečna do nekonečna lze demonstrovat postup, kterým se počítá křížová korelace. Ve výchozím stavu má slider hodnotu posunutí 20, aby byl v následujícím bloku zobrazen konečný výsledek křížové korelace.

V posledním bloku se nachází graf s (částečným) výsledkem křížové korelace mezi prvním a druhým signálem. Částečnost vykresleného výsledku závisí na hodnotě posunutí druhého signálu. Pokud je slider posunutý v záporných číslech, není vykreslen žádný výsledek. První vzorek se vykresluje, při hodnotě slideru 0 – oba vstupní signály se časově překrývají právě v jednom bodě. První vykreslený prvek (s indexem 0) ve výstupním grafu se spočítá jako součin prvků prvního a druhého signálu, které se překrývají (mají totožné indexy). Každý další následující prvek se

počítá jako součet součinů všech prvků prvního a druhého signálu, které se překrývají (mají shodné indexy). Pokud slider posunutí přesáhne hodnotu 19, výsledný graf zobrazuje úplný výsledek křížové korelace signálů.

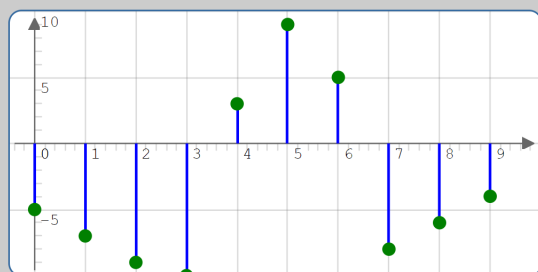
Zadejte první signál:



Vynulovat

Náhodně

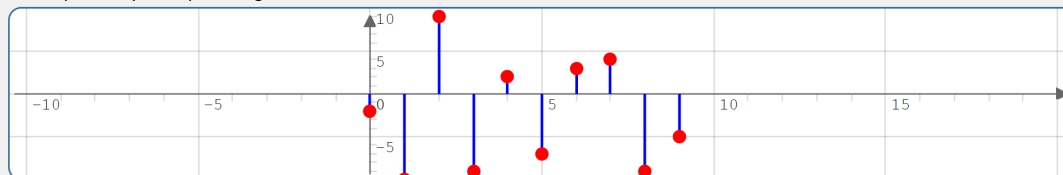
Zadejte druhý signál:



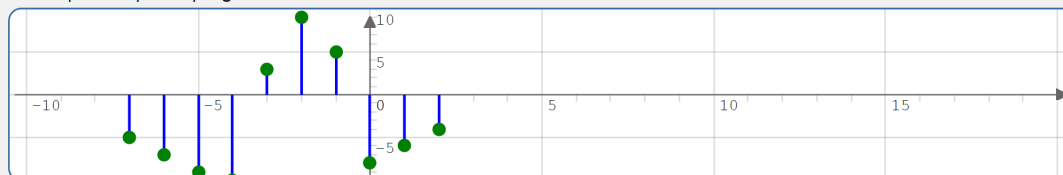
Vynulovat

Náhodně

1.) Zkopírovaný fixní první signál:



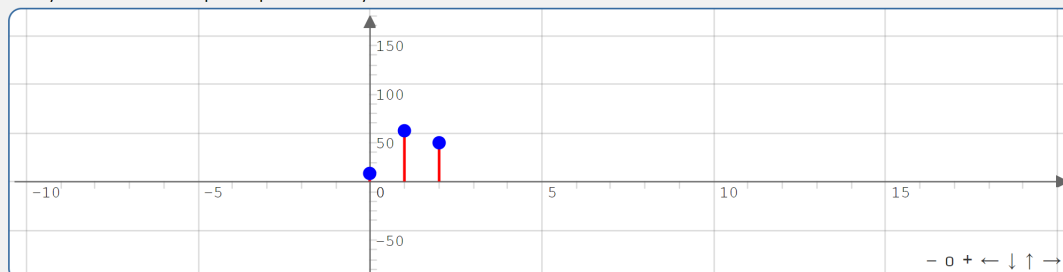
2.) Zkopírovaný druhý signál:



3.) Posunutí druhého signálu o 2:



4.) Výsledek korelace podle posunutí (vyskreseno 3 z 19):



- 0 + ← ↓ ↑ →

Obr. 6.5: Aplet Diskrétní křížová korelace.

## 7 Závěr

V rámci bakalářské práce byla popsána problematika týkající se některých postupů při zpracování signálů. Konkrétně byl text práce zaměřen na záležitosti týkající se zadaných aplikací, které byly v rámci práce vytvořeny.

První čtyři kapitoly obsahují teoretický popis pro jednotlivé aplety, které jsou součástí bakalářské práce. Teoretický popis je doplněn několika grafy, které mají především ilustrativní význam. Čtenář si tak může jednodušeji představit popsanou problematiku.

Pátá kapitola se věnuje vývoji apletů. Jsou zde zmíněné a stručně popsane vybrané programovací jazyky, pomocí kterých webové aplikace vznikaly. Je zde také uveden důvod použití JavaScriptové knihovny JSXGraph. Dále je v kapitole popsáno chování jednotlivých apletů testovaných na různých webových prohlížečích na různých operačních systémech.

V šesté kapitole jsou popsány jednotlivé aplety a jejich chování. Ke každému ovládacímu prvku a výstupním grafům je uveden vystihující komentář.

Praktickým výstupem této bakalářské práce je pět interaktivních apletů určených pro podporu výuky signálů. Jedná se o aplety: Převzorkování diskrétního signálu v poměru racionálního čísla, Střední a efektivní hodnota signálu, Diskrétní lineární konvoluce v 1D, Diskrétní kruhová konvoluce v 1D a Diskrétní křížová korelace. Každý z apletů obsahuje vstupní a výstupní část. Aplety mají podobné grafické rozhraní, které má působit moderním dojmem.



# Literatura

- [1] BRANČÍK, Lubomír. *Elektrotechnika 1*. Brno: VUTIUM, 2004. ISBN 80-214-2607-1.
- [2] BURANSKÝ, Imrich. *HTML a DHTML: hotová řešení*. Brno: Computer Press, 2003. K okamžitému použití (Computer Press). ISBN 80-722-6841-4.
- [3] HAVLÍČEK, Václav, Martin POKORNÝ a Ivan ZEMÁNEK. *Elektrické obvody 1*. Praha: Česká technika – nakladatelství ČVUT, 2005. ISBN 978-80-01-05492-5.
- [4] JAN, Jiří. *Digital signal filtering, analysis and restoration*. London: Institution of Electrical Engineers, 2000. IEE telecommunications series. ISBN 08-529-6760-8.
- [5] JAN, Jiří. *Číslíková filtrace, analýza a restaurace signálů*. Brno: Vysoké učení technické, 1997. ISBN 80-214-0816-2.
- [6] KOSEK, Jiří. *HTML: tvorba dokonalých WWW stránek: podrobný průvodce*. Praha: Grada, 1998. Průvodce (Grada). ISBN 80-716-9608-0.
- [7] LYONS, Richard. *Understanding digital signal processing*. 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2011. ISBN 978-0-13-702741-5.
- [8] MIAO, George. *Signal processing in digital communications*. Boston, MA: Artech House, 2007. ISBN 15-805-3667-0.
- [9] MINKOFF, John. *Signal processing fundamentals and applications for communications and sensing systems*. Boston: Artech House, 2002. ISBN 15-805-3360-4.
- [10] MIŠUREC, Jiří. *Základní metody číslicového zpracování signálů pro integrovanou výuku VUT a VŠB-TUO*. Brno: Vysoké učení technické v Brně, 2014. ISBN 978-80-214-5118-6.
- [11] MITRA, Sanjit Kumar a James KAISER. *Handbook for digital signal processing*. New York: Wiley, 1993. ISBN 04-716-1995-7.
- [12] MUSCIANO, Chuck. *HTML a XHTML: kompletní průvodce*. Praha: Computer Press, 2000. Všechny cesty k informacím. ISBN 80-722-6407-9.
- [13] PEHLIVANIAN, Ara a Don NGUYEN. *JavaScript okamžitě*. Brno: Computer Press, 2014. ISBN 978-80-251-4163-2.

- [14] PÍSEK, Slavoj. *JavaScript: efektní nástroj oživení WWW stránek*. Praha: Grada, 2001. ISBN 80-247-0014-X.
- [15] PROAKIS, John a Dimitris MANOLAKIS. *Digital signal processing: principles, algorithms, and applications*. Upper Saddle River, New Jersey: Prentice Hall, 1996. ISBN 0-13-373762-4.
- [16] RAJMIC, Pavel a Pavel ZÁVIŠKA. *BASS – materiály ke cvičení odborného základu*. Brno, 2017.
- [17] RESIG, John. *JavaScript a Ajax: moderní programování webových aplikací*. Brno: Computer Press, 2007. ISBN 978-80-251-1824-5.
- [18] SMÉKAL, Zdeněk. *Analýza signálu a soustav – BASS*. Brno: Vysoké učení technické v Brně, 2012. ISBN 978-80-214-4453-9.
- [19] SMÉKAL, Zdeněk. *Číslicové zpracování signálů*. Brno: Vysoké učení technické v Brně, 2012. ISBN 978-80-214-4639-7.
- [20] SMÉKAL, Zdeněk. *Systémy a signály: 1D a 2D diskrétní a číslicové zpracování*. Praha: Sdělovací technika, 2013. ISBN 978-80-86645-23-0.
- [21] SMITH, Steven. *Digital signal processing: a practical guide for engineers and scientists*. Boston: Newnes, 2003. ISBN 978-0-7506-7444-7.
- [22] STANÍČEK, Petr. *CSS Kaskádové styly: kompletní průvodce*. Praha: Computer Press, 2003. ISBN 80-722-6872-4.
- [23] UHLÍŘ, Jan a Pavel SOVKA. *Číslicové zpracování signálů*. Praha: Vydavatelství ČVUT, 2002. ISBN 80-01-02613-2.
- [24] VÁCLAVEK, Petr. *JavaScript: hotová řešení*. Brno: Computer Press, 2003. K okamžitému použití (Computer Press). ISBN 80-722-6854-6.
- [25] VLACH, Jan, Zdeněk PRŮŠA a Pavel RAJMIC. Interaktivní applety pro vybrané operace se signály. *Elektrorevue* [online]. 2009 [cit. 2019-05-20]. ISSN 1213-1539. Dostupné z: <<http://www.elektrorevue.cz/cz/download/interaktivni-applety-pro-vybrane-operace-se-signaly-1/>>

# Seznam symbolů, veličin a zkratek

<b>1D</b>	jednorozměrný prostor – One-dimensional space
<b>HTML</b>	hypertextový značkovací jazyk – Hypertext Markup Language
<b>CSS</b>	kaskádové styly – Cascading Style Sheets
<b>GCD</b>	největší společný dělitel – Greatest common divisor
$t$	čas
$n$	index
$s(t)$	analogový signál
$s[n]$	digitální signál
$f$	frekvence
$f_{\text{vz}}$	vzorkovací kmitočet
$T_{\text{vz}}$	vzorkovací perioda
$L$	činitel nadvzorkování
$D$	činitel podvzorkování
$I_{\text{s}}$	střední hodnota
$I_{\text{sa}}$	střední absolutní hodnota
$I_{\text{ef}}$	efektivní hodnota
$I_{\text{m}}$	maximální hodnota
$k_{\text{t}}$	činitel tvaru
$k_{\text{v}}$	činitel výkyvu
$k_{\text{p}}$	činitel plnění
$g(t)$	impulzní odezva analogového systému
$h[n]$	impulzní odezva digitálního systému
$*$	operátor lineární konvoluce
$\circledast$	operátor kruhové konvoluce
$\star$	operátor korelace

## A Obsah přiloženého CD

Přiložené CD obsahuje celkem šest složek a hlavní soubor `index.html`, který slouží jako rozcestník pro spuštění jednotlivých apletů. Ve společné složce `0background` se nachází soubory, díky kterým dostávají jednotlivé aplety jednotné grafické rozhraní a celkovou funkčnost. Nachází se zde důležité soubory `jsxgraphcore.js` a `jsxgraph.css`, které jsou součástí knihovny JSXGraph, díky které jsou v jednotlivých apletech zobrazeny grafy a částečná navigace. Dále je ve složce logo FEKT VUT a fonty Vafle VUT a Vafle Light VUT, které jsou v apletech použity.

Složky jednotlivých apletů obsahují zpravidla tři soubory se zdrojovými kódy. Ke spuštění jednotlivých apletů je určen soubor `index.html`. Funkčnost a interaktivitu dodávají apletům soubory JavaScriptu s koncovkou `.js`. Grafické rozhraní upravují soubory CSS s koncovkou `.css`.

```
/ ..... kořenový adresář přiloženého CD
├── index.html ..... rozcestník
├── 0background ..... společná složka
│   ├── 0style.css
│   ├── fekt.png
│   ├── jsxgraph.css
│   ├── jsxgraphcore.js
│   ├── layout.css
│   ├── vafle.ttf
│   └── vafle-light.ttf
├── 1prevrozkovaniRacionalniCislo ..... aplet Převzorkování v poměru rac. čísla
│   ├── 1script.js
│   ├── 1style.css
│   └── index.html
├── 2stredniEfektivniHodnota ..... aplet Střední a efektivní hodnota signálu
│   ├── 2script.js
│   ├── 2style.css
│   └── index.html
├── 3AlinearniKonvoluce ..... aplet Diskrétní lineární konvoluce v 1D
│   ├── 3Ascript.js
│   ├── 3Astyle.css
│   └── index.html
├── 3BkruhovaKonvoluce ..... aplet Diskrétní kruhová konvoluce v 1D
│   ├── 3Bscript.js
│   ├── 3Bstyle.css
│   └── index.html
├── 4krizovaKorelace ..... aplet Diskrétní křížová korelace
│   ├── 4script.js
│   ├── 4style.css
│   └── index.html
└── MasekBP_apletySignaly.pdf ..... elektronická verze bakalářské práce
```